

Requalificação de uma máquina CNC

JOANA ALEXANDRA CAMBÉ FERREIRA
Setembro de 2024

POLITÉCNICO DO PORTO
INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

Requalificação de uma máquina CNC

Joana Alexandra Cambé Ferreira

Mestrado em Engenharia Electrotécnica e de Computadores
Área de Especialização em Automação e Sistemas



DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA
Instituto Superior de Engenharia do Porto

Setembro, 2024

Esta dissertação satisfaz, parcialmente, os requisitos que constam da Ficha de Unidade Curricular de Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Electrotécnica e de Computadores, Área de Especialização em Automação e Sistemas.

Candidato: Joana Alexandra Cambé Ferreira, N.º 1220143,
1220143@isep.ipp.pt

Orientação Científica: Lino Figueiredo, lbf@isep.ipp.pt

Coorientação Científica: António Meireles, aem@isep.ipp.pt



DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA
Instituto Superior de Engenharia do Porto
Rua Dr. António Bernardino de Almeida, 431, 4200-072 Porto

Setembro, 2024

Dedico este trabalho à minha família, especialmente ao meu companheiro e ao meu filho, pelo amor incondicional, pelo apoio constante, pela força e por acreditarem em mim em todos os momentos, permitindo terminar esta jornada.

Dedico, também, ao meu patrão, que, infelizmente, agora é uma estrelinha no céu, por me ter desafiado a ingressar nesta aventura, passados tantos anos do término da minha Licenciatura. Um muito obrigada por todos os ensinamentos e motivações que me foi dando ao longo do meu percurso profissional.

Agradecimentos

A realização deste projeto só foi possível graças ao apoio e motivação de várias pessoas, assim como do Instituto Superior de Engenharia do Porto (ISEP) e da empresa Padrão Ortopédico, onde o projeto foi realizado, aos quais sou eternamente grata.

Em primeiro lugar, quero agradecer à minha família, nomeadamente ao meu companheiro e filho, por permitirem-me terminar este projeto, pela paciência que tiveram ao longo destes dois anos, abdicando de todas as horas que poderíamos ter passado em família, mas que foram fundamentais para concluir esta fase da minha vida.

Agradeço, igualmente, aos meus orientadores, Eng. Lino Figueiredo e Eng. António Meireles, por aceitarem orientar a minha tese, pelo incentivo e apoio contínuo ao longo deste percurso e durante o meu percurso académico no ISEP.

Também agradeço à empresa Padrão Ortopédico, a minha "casa" profissional desde que terminei a Licenciatura em 2011, por permitirem a realização deste projeto, e pela ajuda e incentivo de todos os meus colegas nesta empresa.

Por fim, agradeço aos meus amigos, que, condicional ou incondicionalmente, me apoiaram, proporcionando o suporte emocional, a alegria e o incentivo necessários nesta fase.

A todos que, de alguma forma, contribuíram para a realização deste trabalho, o meu muito obrigada.

Resumo

As tecnologias de fabrico avançada têm desempenhado um papel crucial na melhoria da qualidade e precisão de dispositivos médicos na área da ortoprotesia. A utilização de máquinas *Computer Numeric Control* (CNC) tem permitido a produção de moldes positivos mais precisos e com melhores acabamentos, reduzindo a necessidade de intervenção humana intensiva e aumentando a eficiência dos processos de fabrico de próteses e ortóteses.

Esta dissertação tem como objetivo a requalificação de uma máquina CNC de três eixos, existente na empresa Padrão Ortopédico, com o intuito de aprimorar a precisão e a eficiência no fabrico dos moldes utilizados no fabrico de dispositivos médicos. A requalificação visa otimizar a integração dos componentes do sistema CNC, sendo o controlo efetuado através de um Arduino Mega 2560, equipado com o *firmware* Grbl-Mega-5X.

O trabalho realizado incluiu a análise e reconfiguração das ligações elétricas, a otimização dos *drivers* e ajustes no *software* nos parâmetros de controlo dos motores, para atingir a precisão necessária. Além disso, foi desenvolvida uma interface gráfica de controlo, através do *software* Grbl Controller, que facilitou a operação da máquina e a sua monitorização em tempo real, permitindo que o processo de fabrico fosse realizado de forma mais eficiente e com menor margem de erro, beneficiando diretamente a qualidade e a personalização dos dispositivos ortopédicos produzidos.

Palavras-Chave: *Computer Numeric Control* (CNC), molde, poliuretano (PU), próteses, ortóteses, *firmware*, *software*, *hardware*, arduino, motores, eixos.

Abstract

Advanced manufacturing technologies have played a crucial role improving the quality and precision of medical devices in the field of orthotics and prosthetics. The use of *Computer Numeric Control* (CNC) machines has enabled the production of more precise positive molds with better finishes, reducing the need for intensive human intervention and increasing the efficiency of the manufacturing processes for prostheses and orthoses.

This dissertation aims to requalify a three-axis CNC machine, currently in use at Padrão Ortopédico, with the goal of enhancing the precision and efficiency in the manufacturing of molds used for medical devices. The requalification seeks to optimize the integration of the CNC system components, with control being managed by an Arduino Mega 2560 equipped with the Grbl-Mega-5X firmware.

The work carried out included the analysis and reconfiguration of the electrical connections, optimization of the drivers and adjustments on the motor control parameters, to achieve the required precision. Additionally, a graphical user interface was developed using Grbl Controller software, helping on the operation of the machine and real-time monitoring. This allows the manufacturing process to be carried out more efficiently and with a lower margin of error, directly benefiting the quality and customization of the orthopedic devices produced.

Keywords: CNC, cast, PU, prosthesis, orthosis, *firmware*, *software*, *hardware*, arduino, motors, axis.

Índice

Lista de Figuras	xi
Lista de Tabelas	xvii
Lista de Acrónimos	xix
Lista de Símbolos	xxi
1 Introdução	1
1.1 Contextualização	2
1.2 Definição do Problema	3
1.2.1 Objetivos	4
1.2.2 Resultados esperados	4
1.3 Plano de Trabalho	5
1.4 Organização da Dissertação	5
2 Estado da arte	7
2.1 Introdução à Ortoprotesia	7
2.1.1 Próteses ou dispositivos protésicos	9
2.1.2 Ortóteses ou dispositivos ortésicos	11
2.2 Fabrico de próteses e ortóteses	14
2.2.1 Processo tradicional de fabrico de moldes positivos	15
Toma de medidas do membro ou segmento corporal	15
Molde negativo e positivo (3D) do membro ou segmento corporal	16
Retificação do molde positivo de acordo com os parâmetros	
desejados	18
2.2.2 Processo digital de fabrico de moldes positivos	19
Toma de medidas do membro ou segmento corporal	20
Molde positivo tridimensional do membro ou segmento corporal	20
Retificação do molde positivo de acordo com parâmetros de-	
sejados	22
Produção do molde positivo de acordo com os parâmetros de-	
sejados	23
2.2.3 Fabrico da ortótese ou prótese através do molde positivo	23

2.3	Conclusão do Capítulo 2	26
3	Tecnologia CNC e maquinagem de materiais	27
3.1	Introdução ao <i>Computer Numeric Control</i> (CNC)	27
3.1.1	Sistema de coordenadas nas máquinas CNC	30
3.2	Composição das máquinas-ferramenta CNC	32
3.2.1	Estrutura	33
3.2.2	Acionamento rotativo	34
	Acionamento dos eixos	34
	Acionamento da árvore	34
	Servomotores	35
	Motores de passo	37
3.2.3	Acionamento linear	38
	Atuadores piezoelétricos	38
	Motores elétricos lineares	38
	Atuadores hidráulicos	39
	Atuadores pneumáticos	39
3.2.4	Sistema de controlo e sensores	39
	Codificadores	39
	<i>Resolver</i>	42
3.2.5	Sistema de transmissão e conversão de movimento	42
	Guias lineares	42
	Sistemas de transmissão de movimento	44
	Sistemas de conversão de movimento	46
3.2.6	Sistema de acoplamento	47
	<i>Drivers</i> dos motores	48
	Fonte de alimentação	49
3.3	Componentes do sistema CNC	50
3.3.1	<i>Man-Machine Interface</i> (MMI)	52
3.3.2	<i>Numeric Control Kernel</i> (NCK)	53
3.3.3	<i>Programmable Logic Control</i> (PLC)	53
3.4	Utilização de máquinas CNC em ortoprotesia	54
3.4.1	Máquinas fresadoras	56
3.4.2	Máquinas torneadoras	57
3.5	Tecnologia existente no mercado	59
3.5.1	Máquinas-ferramenta CNC	60
	Rodin4D Model SX	60
	Vulcan TWINCAM	62
3.5.2	Controladores e MMI para máquinas CNC	63
3.5.3	<i>Software</i> CAM para gerar G-code	67

3.5.4	<i>Software e firmware</i> para controlo de máquinas CNC	68
3.6	Maquinagem de objetos	71
3.6.1	Fresagem	71
3.6.2	Torno	73
3.7	Conclusão do capítulo 3	74
4	Estudo da máquina CNC "QRC"	75
4.1	Estrutura e arquitetura do sistema CNC "QRC"	75
4.1.1	Estrutura mecânica da máquina CNC "QRC"	77
	Eixos de movimento	78
	Mesa de trabalho, peça de fixação e ferramenta de corte	82
4.1.2	Sistema de transmissão de movimento	83
	Motores utilizados	83
	<i>Spindle</i> elétrico	85
4.1.3	Sistema de controlo	86
	Microprocessador Arduino Mega 2560 Rev3	86
	<i>Drivers</i> dos motores	87
	Fonte tensão do <i>spindle</i>	88
	Fonte de alimentação	89
	<i>Firmware</i> e controlador Grbl	89
4.2	Funcionamento da máquina CNC "QRC"	91
4.2.1	Controlador, <i>software</i> e <i>firmware</i>	92
4.2.2	<i>Man Machine Interface</i> da máquina CNC "QRC"	94
4.3	Conclusão do Capítulo 4	95
5	Desenvolvimento do projeto	97
5.1	Ligações elétricas do <i>hardware</i> do sistema	97
5.1.1	Ligação entre os <i>drivers</i> e os motores	99
	Ligação dos motores NEMA 23 aos <i>drivers</i> DM542T	99
	Ligação dos motores NEMA 34 aos <i>drivers</i> DM860I	100
	Retificação das ligações entre os motores e os <i>drivers</i>	101
	Ajustes aos <i>drivers</i>	102
5.1.2	Ligação entre as fontes de alimentação e os <i>drivers</i>	103
	Retificação das ligações entre as fontes de alimentação e os <i>drivers</i>	105
5.1.3	Ligações entre o controlador e os <i>drivers</i>	106
	Ligação dos <i>drivers</i> do eixo A e Z ao controlador	107
	Ligação dos <i>drivers</i> do eixo X ao controlador	108
5.1.4	Ligações entre o controlador, o <i>firmware</i> e o <i>software</i> de controlo	108
	Comunicação entre o <i>firmware</i> Grbl e o Arduino Mega 2560	109
	Comunicação entre o <i>firmware</i> Grbl e o Grbl Controller	110

5.2	Testes de validação e ajustes finais	112
5.2.1	Testes aos movimentos dos motores	113
	Configuração dos pinos para testar os motores NEMA 23 . . .	113
	Configuração dos pinos para testar os motores NEMA 34 . . .	113
5.2.2	Testes aos movimentos dos eixos	114
	Reconfiguração dos pinos dos eixos da máquina CNC "QRC" .	115
	Configuração dos limites máximos dos eixos	116
5.2.3	Testes realizados aos motores	117
	Testes aos micropassos dos motores	117
	Testes à velocidade dos motores	118
	Testes à aceleração dos motores	119
5.3	Fabrico de moldes na máquina CNC "QRC"	120
5.3.1	Digitalizar e retificar o molde positivo 3D	120
5.3.2	Orientar o molde positivo com as coordenadas da máquina "QRC"	122
5.3.3	Gerar G-code	123
	Definição dos parâmetros da máquina e do pós-processador .	124
	Definir parâmetros da ferramenta de corte	124
	Configurar parâmetros de maquinagem e operação da máquina	125
	Simulação do tempo de maquinagem e da peça final	129
	Exportar o G-code	130
5.3.4	Preparar bloco de PU para maquinar	131
5.3.5	Preparar máquina para a maquinagem	132
5.3.6	Maquinagem do bloco de PU	135
5.4	Resultados obtidos com a maquinagem	135
5.5	Limitações encontradas na máquina "QRC"	137
5.6	Plano de manutenção e melhorias futuras para o sistema	137
5.7	Conclusão do Capítulo 5	139
6	Conclusões	141
6.1	Considerações finais	141
6.2	Trabalho Futuro	142
	Referências	145
	Anexo A Desenhos CAD	153
A.1	Desenho CAD da peça de fixação	153
A.2	Desenho técnico da peça de fixação	154

Anexo B Especificações do sistema CNC "QRC"	155
B.1 Dimensões dos motores	155
B.2 <i>Pinout</i> inversor do <i>Spindle</i>	156
B.3 <i>Pinout</i> Arduino Mega 2560 Rev3	157
B.4 Dimensões da fonte de alimentação	158
B.5 Circuito elétrico do sistema CNC "QRC"	159
B.6 Resolução dos <i>drivers</i>	160
B.7 Dimensões dos <i>drivers</i>	161
Anexo C G-code e comandos do sistema Grbl	163
C.1 G-code suportado pelo Grbl	163
C.2 Comandos próprios do sistema Grbl	165
Anexo D Código utilizado no <i>software</i> Arduino IDE	167
D.1 Código utilizado para testar os motores	167
D.1.1 Código para testar motores NEMA 23	167
D.1.2 Código para testar motores NEMA 34	168
D.2 Mapeamento de pinos do Arduino para o Grbl	169
D.2.1 Ficheiro <code>cpu_map.h</code>	169
D.2.2 Ficheiro <code>config.h</code>	171
D.2.3 Ficheiro <code>stepper.h</code>	174
D.2.4 Ficheiro <code>spindle_control.h</code>	175
D.2.5 Ficheiro <code>spindle_control.c</code>	175
D.2.6 Ficheiro <code>probe.h</code>	176
D.2.7 Ficheiro <code>coolant_control.h</code>	177
D.2.8 Ficheiro <code>coolant_control.c</code>	178
D.2.9 Ficheiro <code>limits.h</code>	178

Lista de Figuras

2.1	Amputações da extremidade inferior (a) e superior (b) [3]	8
2.2	Exemplos de próteses desenhadas para amputações do membro inferior: transtibial (a) (c) e transfemural (b) [5]	9
2.3	Exemplos de próteses desenhadas para amputações do membro superior: transradial (b) (c) e transumeral (a) [5]	10
2.4	Exemplos de ortóteses desenhadas para o membro inferior: tornozelo-pé (a), joelho (b) e joelho-tornozelo-pé (c) [7] [8]	11
2.5	Exemplos de ortóteses desenhadas para o membro superior: mão-punho (a) (b) (c) [7] [9]	12
2.6	Exemplos de ortóteses desenhadas para a coluna (a) (b) [7] [10]	12
2.7	Ligaduras gessadas (a) [12] e de resina (b) [13]	15
2.8	Exemplo de um formulário de medidas para o fabrico de uma ortótese para o membro inferior [14]	16
2.9	Procedimentos realizados aquando da toma do molde negativo, através do método tradicional [11]	17
2.10	Retificação do molde positivo, através do método tradicional [11]	18
2.11	Exemplo de um <i>scanner</i> 3D, para obtenção de molde positivo [15]	19
2.12	[16]	19
2.13	Exemplo de uma máquina de fresar CNC (a) e a obtenção de um molde positivo (b) através da mesma [11]	20
2.14	[15]	21
2.15	[19]	22
2.16	Processos possíveis através do <i>software</i> de modelação [20]	23
2.17	Termomoldagem de uma ortótese para o membro inferior [11]	24
2.18	Moldagem por laminação de um encaixe de uma prótese para o membro inferior [21]	25
2.19	Obtenção de um encaixe de prótese para o membro inferior, recorrendo ao fabrico aditivo [22]	25
2.20	Exemplo de um par de ortóteses (a) e de uma prótese (b) para o membro inferior acabadas [6]	26
3.1	Arquitetura de uma máquina-ferramenta NC e respetivo fluxo de trabalho	28

3.2	Exemplo de sistema de coordenadas de uma máquina CNC [24]	30
3.3	Regra da mão direita [25]	31
3.4	Exemplo de uma máquina de fresagem CNC [1]	33
3.5	Exemplo de um mecanismo de acionamento direto dos eixos de uma máquina-ferramenta [23]	34
3.6	Exemplo de um mecanismo de acionamento da árvore de uma máquina-ferramenta [23]	35
3.7	Tipos de servomotores: DC (a), AC síncrono (b) e AC de indução (c) [23]	35
3.8	Exemplo de motores de passo [30]	37
3.9	Esquema dos codificadores do tipo ótico (a) e magnético (b) [31]	40
3.10	Esquema do funcionamento de um codificador incremental [32]	41
3.11	Esquema do funcionamento de um codificador absoluto [32]	42
3.12	Exemplos de guias de deslizamento: prismática (a) e rabo de andorinha (b) [28]	43
3.13	Exemplos de guias de elementos rolantes: rolo (a) e esfera (b) [28]	43
3.14	Exemplo de um trem de engrenagem [30]	44
3.15	Exemplo de um parafuso sem-fim [30]	44
3.16	Exemplo de um pinhão-cremalheira [30]	45
3.17	Exemplo de uma correia (a) e polia sincronizada (b) [30]	45
3.18	Exemplo de uma corrente de transmissão [30]	46
3.19	Exemplo de um fuso trapezoidal [30]	46
3.20	Exemplo de um fuso de esferas recirculantes [30]	47
3.21	Exemplo de um fuso de rolos [30]	47
3.22	Exemplos de acoplamentos: rígido (a) e flexíveis (b) e (c) [30]	48
3.23	Exemplo de um motor de passo e respetivo <i>driver</i> [30]	48
3.24	Exemplo de uma fonte de alimentação [30]	49
3.25	Relação entre os sistemas CNC [28]	50
3.26	Componentes de um sistema CNC [23]	51
3.27	Exemplo de uma unidade MMI, que mostra algumas funções programáveis [34]	52
3.28	Arquitetura e função do sistema PLC [23]	54
3.29	Exemplo de uma máquina fresadora horizontal [37]	57
3.30	Exemplo de um torno vertical CNC [16]	58
3.31	Máquina CNC Model SX (a) e <i>software</i> CAD/CAM Neo CAM, desenvolvidos pela empresa Rodin4D (b) [38]	60
3.32	Máquina CNC Vulcan TWINCAM, da empresa Sensormedica [39]	62
3.33	Exemplo de <i>kits retrofit</i> CNC da Siemens (a) e Acu-Rite (b) [42]	65
3.34	Exemplo de <i>kit</i> CNC acessível através da AliExpress [43]	65
3.35	Simulação da maquinagem de uma peça no <i>software</i> MasterCam [45]	68

3.36	Simulação de uma peça maquinada no <i>software</i> DeskProto [46]	68
3.37	Visualização de G-code e maquinagem no <i>software</i> UGS [47]	69
3.38	<i>Software</i> MACH4 [48]	70
3.39	Esquema de maquinagem de objetos pelo processo de fresar: cilíndrica (a) e frontal (b) [53]	72
3.40	Esquema de maquinagem de objetos pelo processo de tornear [53]	73
4.1	Máquina CNC "QRC", disponível na empresa Padrão Ortopédico	76
4.2	Estrutura da máquina "QRC", desenhada através de <i>software</i> CAD	78
4.3	Representação dos eixos de movimento X, Z e A, da máquina "QRC"	79
4.4	Área de trabalho da máquina "QRC"	80
4.5	Representação do motor e mecanismo de transmissão de movimento relativo ao eixo A, na parte superior da área de trabalho	80
4.6	Representação dos componentes constituintes do eixo X	81
4.7	Representação dos componentes constituintes do eixo Z	82
4.8	Representação do motor e mecanismo de transmissão de movimento relativo ao eixo A	82
4.9	Peça de fixação do bloco responsável pela ligação à placa rotativa	83
4.10	Exemplo da ferramenta de corte utilizada na máquina "QRC" [55]	83
4.11	Exemplos de motores NEMA 23 (a) e NEMA 34 (b) utilizados na máquina [56] [57]	84
4.12	Motor NEMA 23, acoplado na parte inferior do prato de rotação da máquina "QRC"	84
4.13	<i>Spindle</i> elétrico encontrado na máquina "QRC" [58]	85
4.14	Área de controlo (a) e componentes eletrónicos (b), da máquina "QRC"	86
4.15	Microprocessador Arduino Mega 2560 REV3 [59]	87
4.16	<i>Drivers</i> utilizados na máquina "QRC": DM860I (a) e DM542T (b) [60] [61]	88
4.17	Inversor do <i>spindle</i> YL620-A [62]	89
4.18	Fonte de alimentação S-350-60 utilizada na máquina [63]	90
4.19	Logótipo da versão 5/6 eixos Grbl-Mega-5X [64]	90
4.20	Página de arranque do <i>software</i> Grbl Controller	91
4.21	Diagrama ilustrativo das ligações estabelecidas entre o <i>software</i> , <i>firmware</i> e controlador e os sistemas de transmissão de movimento e mecânicos	92
5.1	Diagrama de blocos da arquitetura do sistema CNC "QRC"	98
5.2	Diagrama das ligações possíveis ao motor NEMA 23 [65]	99
5.3	Diagrama das ligações entre o <i>driver</i> DM542T e o NEMA 23 [65]	100
5.4	Diagrama das ligações possíveis ao motor NEMA 34 [66]	100
5.5	Diagrama das ligações entre o <i>driver</i> DM860I e o NEMA 34 [66]	101

5.6	Ligações entre os <i>drivers</i> DM860I e duas fontes de alimentação S-350-60 [60] [63]	104
5.7	Ligações entre os <i>drivers</i> DM542T e uma fonte de alimentação S-350-60 [61] [63]	105
5.8	Mensagem de início enviada pelo Grbl, visível no <i>Serial Monitor</i> do Arduino IDE	110
5.9	Parâmetros do Grbl configurados no Arduino, visíveis no <i>Serial Monitor</i> do Arduino IDE	110
5.10	Mensagem de início enviada pelo Grbl, através da interface gráfica do Grbl Controller	112
5.11	Comandos enviados para testar o movimento do eixo X	115
5.12	Nova mensagem de início enviada pelo Grbl, visível no <i>Serial Monitor</i> do Arduino IDE	115
5.13	Novos parâmetros do Grbl configurados no Arduino, visíveis no <i>Serial Monitor</i> do Arduino IDE	116
5.14	Comandos \$130, \$132 e \$133, referentes aos limites dos eixos	117
5.15	Comandos \$100, \$102 e \$103, referentes aos micropassos dos motores	118
5.16	Comandos \$110, \$112 e \$113, referentes à velocidade dos motores	119
5.17	Comandos \$120, \$122 e \$123, referentes à aceleração dos motores	120
5.18	Digitalização de um membro residual com recurso a um <i>scanner</i> 3D	121
5.19	<i>software</i> CAD, OrtenShape para retificação do molde positivo	121
5.20	Alinhamento do molde do membro digitalizado, conforme os eixos de coordenadas da máquina	122
5.21	Alinhamento dos eixos para a maquinaagem, respeitando as características da máquina	123
5.22	Configuração da máquina "QRC" (a) e habilitação do eixo A (b), no <i>software</i> DeskProto	124
5.23	Configuração da ferramenta de corte, no <i>software</i> DeskProto	125
5.24	Configurações gerais dos parâmetros da peça (a) e das dimensões do bloco a maquinar (b), no <i>software</i> DeskProto	126
5.25	Configuração dos valores de precisão e velocidade (a) e estratégia de fresagem (b) para a primeira operação, no <i>software</i> DeskProto	127
5.26	Configuração de corte e profundidade (a) e movimento (b) da ferramenta, para a primeira operação, no <i>software</i> DeskProto	127
5.27	Configuração dos valores de precisão e velocidade, para a segunda operação, no <i>software</i> DeskProto	128
5.28	Configuração de corte e profundidade da ferramenta, para a segunda operação, no <i>software</i> DeskProto	129
5.29	Simulação dos <i>paths</i> que a ferramenta de corte realiza na operação <i>first milling</i> (a) e na operação <i>finishing</i> (b), no <i>software</i> DeskProto	129

5.30	Estimação do tempo de fresagem para as duas operações, no <i>software</i> DeskProto	130
5.31	Simulação do bloco de PU (a) e do molde positivo (b), no <i>software</i> DeskProto	130
5.32	Partes iniciais do G-code da operação <i>first milling</i> (a) e <i>finishing</i> (b)	131
5.33	Bloco de PU fixado à peça de fixação (a) e colocado na máquina CNC no eixo A (b)	132
5.34	Interruptores ON/OFF da máquina "QRC"	132
5.35	Posição inicial da ferramenta de corte aquando da maquinagem do bloco de PU	134
5.36	Controlador do <i>spindle</i>	134
5.37	Bloco de PU maquinado	136
A.1	Desenhos CAD da peça de fixação	153
A.2	Desenho técnico da peça de fixação	154
B.1	Dimensões do motor NEMA 23 [56]	155
B.2	Dimensões do motor NEMA 34 [57]	156
B.3	Esquema de ligações possíveis ao inversor do <i>Spindle</i> YL620-A [62] .	156
B.4	<i>Pinout</i> Arduino Mega 2560 Rev3 [71]	157
B.5	Dimensões da fonte de alimentação S-350-60 [63]	158
B.6	Circuito elétrico do sistema CNC "QRC"	159
B.7	Dimensões do <i>driver</i> DM860I [60]	161
B.8	Dimensões do <i>driver</i> DM542T [61]	161

Lista de Tabelas

1.1	Calendarização das tarefas planeadas para Tese/Dissertação (TEDI)	5
2.1	Classificação do tipo de amputação e próteses do membro inferior (a) e do membro superior (b) [3] [4]	10
2.2	Classificação do tipo de ortótese para o membro inferior [3] [4]	13
2.3	Classificação do tipo de ortótese para o membro inferior [3] [4]	13
2.4	Classificação do tipo de ortótese para a coluna [3] [4]	13
2.5	Passos essenciais para a aplicação de dispositivos médicos [2]	15
3.1	Endereços de comando numérico em G-code [26]	32
3.2	Resumo das vantagens, desvantagens e características de cada motor [23]	37
3.3	Características e especificações da máquina CNC Model SX da empresa Rodin4D [38]	61
3.4	Características e especificações da máquina CNC Vulcan TWINCAM da empresa Sensor Medica [39]	63
4.1	Componentes da máquina "QRC"	77
B.1	Tabela dos micropassos e passos/revolução do <i>driver</i> DM860I [72]	160
B.2	Tabela dos micropassos e passos/revolução do <i>driver</i> DM542T [73]	160
C.1	G-code suportado pelo <i>firmware</i> Grbl: comandos modais [74] [75]	164
C.2	G-code suportado pelo <i>firmware</i> Grbl: comandos não modais [74] [75]	165
C.3	Sistema de comandos \$ específico do Grbl [74] [75]	165
C.4	Sistema de comandos \$ ASCII, específico do Grbl [74] [75]	166

Lista de Acrónimos

3D	<i>three dimension</i>
AC	corrente alternada
AP	ântero-posterior
ASCII	<i>American Standard Code for Information Interchange</i>
CAD	<i>Computer Aided Design</i>
CAD/CAM	<i>Computer Aided Design/Computer Aided Manufacturing</i>
CAM	<i>Computer Aided Manufacturing</i>
CNC	<i>Computer Numeric Control</i>
CPU	<i>Central Processing Unit</i>
D/A	<i>analógico/digital</i>
DC	corrente contínua
EEPROM	<i>Electrically Erasable Programmable Read-Only Memory</i>
EUA	Estados Unidos da América
EVA	acetato de etileno-vinil
G-code	<i>Geometric Code</i>
GND	<i>Ground</i>
I/O	<i>input-output</i>
IDE	<i>Integrated Development Environment</i>
ISEP	Instituto Superior de Engenharia do Porto
ISO	<i>International Society Organization</i>
LCD	<i>Liquid Crystal Display</i>
LED	<i>light-emitting diode</i>

MCU	<i>microcontroller unit</i>
MEEC	Mestrado em Engenharia Electrotécnica e Computadores
MIT	Massachusetts Institute of Technology
ML	médio-lateral
MMI	<i>Man-Machine Interface</i>
NC	<i>Numeric Control</i>
NCK	<i>Numerical Control Kernel</i>
NEMA	<i>National Electrical Manufacturers Association</i>
PC	computador pessoal
PE	polietileno
PID	<i>Proportional-Integral-Derivative</i>
PLC	<i>Programmable Logic Control</i>
PP	polipropileno
PU	poliuretano
PVA	álcool polivinílico
RAM	<i>random-access memory</i>
ROM	<i>read-only memory</i>
SD	<i>Secure Digital</i>
SO	Sistema Operativo
SRAM	<i>Static Random Access Memory</i>
TEDI	Tese/Dissertação
UGS	Universal Gcode Sender
USB	<i>Universal Serial Bus</i>
VDC	voltagem corrente contínua

Lista de Símbolos

Símbolo	Descrição	Unidades
ρ	densidade	kg/m ³
F	força	N
ν	frequência	Hz
I	corrente	A
m	massa	kg
P	potência	W
t	tempo	s
V	tensão	v
x	deslocamento	m
θ	deslocamento angular	rad
ω	velocidade angular	rad s ⁻¹

Capítulo 1

Introdução

O principal objetivo desta dissertação é requalificar uma máquina CNC existente na empresa Padrão Ortopédico, com o propósito de aperfeiçoar a sua interface de utilizador, melhorando a eficiência e a precisão no processo de produção de moldes para o fabrico de próteses e ortóteses. A requalificação visa otimizar a utilização e desempenho da máquina, promovendo um ambiente de produção mais ágil e eficaz para a fabricação de dispositivos médicos personalizados.

Este projeto surgiu como uma oportunidade de aliar a experiência profissional ao conhecimento teórico adquirido durante o Mestrado em Engenharia Electrotécnica e Computadores (MEEC), proporcionando uma aplicação prática e relevante que contribui diretamente para a melhoria dos processos na empresa. A integração dos conceitos aprendidos ao longo do curso, como controlo de sistemas, automação e programação de máquinas CNC, foi fundamental para desenvolver uma solução que vai de encontro às necessidades específicas do contexto industrial e médico.

Este capítulo apresenta a introdução à temática do projeto, contextualizando o problema identificado e a solução proposta. O enquadramento do projeto na área de desenvolvimento é abordado, destacando a relevância da aplicação das tecnologias de fabrico digital na produção de dispositivos ortopédicos. Além disso, são discutidos a metodologia utilizada e o cronograma estabelecido para a execução da dissertação. Por fim, é realizada uma breve descrição do conteúdo abordado em cada um dos capítulos deste documento, facilitando a leitura e a compreensão do trabalho desenvolvido.

1.1 Contextualização

A área da Ortoprotesia tem testemunhado uma evolução significativa nos últimos anos, impulsionada pela procura de soluções mais eficazes e personalizadas no fabrico de dispositivos médicos. Com o avanço das tecnologias de produção digital, a produção aditiva e o recurso a máquinas CNC, surgem como aliados para alcançar níveis de precisão, qualidade e eficiência, mais difíceis de atingir até ao momento através dos métodos tradicionais.

A maquinação CNC permite a produção de moldes positivos dos membros e componentes ortoprotésicos com uma precisão e repetibilidade elevadas, garantindo um ajuste perfeito e maior conforto para o utilizador final. Estas máquinas automatizadas utilizam *software* de última geração para controlar ferramentas de corte e maquinagem em múltiplos eixos, o que possibilita a criação de geometrias complexas e superfícies suaves, essenciais para o fabrico de dispositivos customizados.

O uso da tecnologia CNC na ortoprotesia não reduz apenas o tempo de fabricação mas minimiza, igualmente, a necessidade de intervenção humana intensiva, permitindo uma produção mais rápida e eficiente de próteses e ortóteses personalizadas. Este avanço tecnológico torna-se especialmente relevante num contexto onde a procura por dispositivos feitos por medida cresce exponencialmente, impulsionada pela necessidade de melhorar a qualidade de vida dos utentes.

Para acompanhar esta evolução, as empresas do setor procuram cada vez mais atualizar-se no sentido de integrar soluções digitais nas empresas, que combinam a flexibilidade e a precisão das máquinas CNC com ferramentas de *Computer Aided Design/Computer Aided Manufacturing* (CAD/CAM). Esta combinação possibilita uma otimização dos processos de *design* e fabrico, reduzindo o desperdício de material e os custos operacionais. Além disso, a capacidade de adaptar rapidamente os processos de produção a diferentes materiais e *designs* faz das máquinas CNC uma escolha ideal para a fabricação de dispositivos ortopédicos inovadores e personalizados.

É possível perceber que a integração das tecnologias de maquinação CNC na área da ortoprotesia representam um avanço significativo na capacidade de resposta às necessidades individuais dos utentes, permitindo um fabrico mais ágil, preciso e economicamente viável. Este paradigma de produção digital tem se refletido nesta indústria, proporcionando às empresas uma vantagem competitiva e aos utentes dispositivos de apoio de alta qualidade, adaptados às suas especificidades.

1.2 Definição do Problema

A indústria Ortoprotésica, como referido, encontra-se em constante evolução tecnológica pela grande demanda no fabrico de produtos de apoio personalizados que proporcionem maior conforto, precisão e funcionalidade aos seus utilizadores. No entanto, um dos grandes desafios encontrados nesta indústria é a produção desses dispositivos envolver o fabrico de moldes positivos, muitas vezes de dimensão média a grande. Pelo método tradicional de fabrico, estes moldes obtêm-se recorrendo ao gesso, que, muitas vezes, pode pesar dezenas de quilogramas, dificultando o seu manuseamento. Além deste processo ser demorado, ao requerer a tiragem do molde com ligaduras de gesso e posterior preenchimento das mesmas com gesso, o material é frágil e rígido, o que faz com que seja complicado realizar modificações nas suas superfícies, tornando o produto final, muitas vezes inadequado, com pouca precisão e com tempo e custos de produção elevados.

Com o intuito de aprimorar o processo de obtenção de moldes positivos para o fabrico de dispositivos ortoprotésicos, recorre-se, cada vez mais, a máquinas CNC, que maquinam, na maioria das vezes, os moldes em poliuretano (PU). Com a utilização deste tipo de tecnologia, existe um enorme potencial em contornar-se as limitações encontradas pelo método tradicional, pois estas máquinas têm uma grande capacidade de execução de operações de maquinagem complexas com alta precisão e com grande repetibilidade [1].

Apesar de se conseguir obter moldes positivos mais leves, e por isso, mais facilmente manuseáveis, a adaptação e otimização das máquinas CNC para este tipo de fabrico apresenta desafios técnicos e de operação específicos que têm de ser estudados. Assim, os principais problemas identificados para este projeto são:

- Redução dos tempos de produção de moldes positivos, com recurso à máquina CNC e a blocos de PU, face ao método tradicional;
- Obtenção de precisão e qualidade nos moldes positivos de PU, através do método digital, resultando em dispositivos que se ajustam perfeitamente ao utilizador;
- Custos associados à requalificação da máquina CNC;
- Custos associados ao desperdício de material (PU);
- Adaptação da tecnologia CNC à área da Ortoprotesia;
- Interface homem-máquina intuitiva para fácil manuseamento por parte do utilizador.

1.2.1 Objetivos

O objetivo principal deste projeto prende-se com o estudo da máquina CNC presente na empresa Padrão Ortopédico, bem como das suas potencialidades para o fabrico de moldes positivos em PU. Além disso, pretende-se explorar e desenvolver algumas soluções de requalificação para mesma, de modo a torná-la mais eficiente, precisa, e, mais facilmente manuseável. Assim, os principais objetivos prendem-se com:

- Analisar os diversos componentes da máquina CNC, ao nível do *hardware* e *software*;
- Identificar os componentes passíveis de melhoria para alcançar maior qualidade e redução no tempo de produção dos moldes positivos;
- Compreender os requisitos técnicos e cuidados necessários na maquinagem do PU em máquinas CNC;
- Desenvolver e testar os procedimentos e configurações de *software* mais adequados para a fresagem dos moldes, garantindo a sua precisão e qualidade;
- Avaliar o impacto das requalificações neste tipo de máquinas, na redução do tempo de produção e dos custos associados;
- Desenvolver uma interface gráfica mais intuitiva e de fácil utilização;
- Propor um modelo de integração entre *softwares* CAD/CAM e a máquina CNC para otimizar o processo de fabricação de dispositivos ortoprotésicos.

1.2.2 Resultados esperados

Com a realização deste projeto e a definição do problema do mesmo, são esperados resultados práticos com a implementação das melhorias propostas. Com base nos objetivos previamente estabelecidos, pretende-se alcançar uma melhoria significativa na eficiência da produção dos moldes positivos, bem como uma otimização dos processos de maquinagem e uma integração mais eficaz dos sistemas CAD/CAM com a máquina CNC. Assim, os resultados esperados são:

- Melhoria da qualidade dos moldes positivos, com o aumento da precisão e consequente melhoria do fabrico dos dispositivos ortoprotésicos;
- Redução do tempo de produção dos moldes, levando a um aumento da produtividade da máquina e do operador;
- Diminuição dos custos operacionais, devido à menor taxa de desgaste dos componentes da máquina;

- Desenvolvimento de uma interface do utilizador mais intuitiva, com uma curva de aprendizagem menor, permitindo, igualmente, a redução de erros de produção;
- Otimização do fluxo de trabalho, desde a modelação do molde positivo 3D até à sua produção, integrando eficazmente os sistemas CAD/CAM e CNC;
- Criar documentos e procedimentos padrão de modo a garantir a consistência e qualidade da produção.

1.3 Plano de Trabalho

Para ser possível uma melhor organização do trabalho e estabelecimento de metas e objetivos no decorrer do projeto, foi elaborado um calendário de trabalho, como se pode observar pela **Tabela 1.1**. Na mesma, são apresentadas as tarefas que se esperam desenvolver e alcançar, no decorrer do projeto, bem como o tempo que se espera despendar em cada uma das mesmas.

Tabela 1.1: Calendarização das tarefas planeadas para Tese/Dissertação (TEDI)

Tarefas	Fevereiro				Março				Abril				Maio				Junho				Julho				Agosto				Setembro			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Definição do tema e proposta	█	█																														
Introdução ao tema do projeto			█	█	█	█																										
Definição dos objetivos do projeto			█	█																												
Pesquisa relativa ao estado de arte					█	█	█	█	█	█	█	█																				
Estudo sobre máquinas CNC									█	█	█	█	█	█	█	█	█	█	█	█												
Análise da máquina em estudo																	█	█	█	█	█	█	█	█								
Testes e aplicação do projeto																					█	█	█	█	█	█	█	█				
Resultados obtidos e conclusão																									█	█	█	█	█	█	█	█
Redação da dissertação																																

1.4 Organização da Dissertação

A presente dissertação encontra-se dividida em seis capítulos, onde o capítulo um, ”**Introdução**”, apresenta o tema do projeto bem como a problemática envolvida na sua execução. Também são enunciados os objetivos que se pretendem alcançar com a realização do projeto e a metodologia utilizada para o efeito.

No segundo capítulo, ”**Estado da arte**”, são introduzidos conceitos fundamentais relacionados à área da Ortoprotesia, abordando os diferentes tipos de próteses e ortóteses, assim como os processos necessários para a obtenção destes dispositivos médicos.

O terceiro capítulo, ”**Máquinas CNC e maquinagem**”, trata de temas fundamentais para a compreensão do projeto desenvolvido. São definidos conceitos

relacionados com as máquinas CNC e apresentadas tecnologias já existentes no mercado para estes tipos de equipamento.

No quarto capítulo, "**Estudo da máquina CNC "QRC"**", é realizada uma análise detalhada da máquina CNC "QRC" enumerando os diferentes componentes de *hardware* e *software*, bem como as ligações estabelecidas entre os mesmos.

O quinto capítulo, "**Desenvolvimento do projeto**", descreve as etapas necessárias para garantir o correto funcionamento da máquina CNC, detalhando as fases fundamentais para a obtenção de um molde positivo.

Por fim, no capítulo seis, "**Conclusão**", são apresentados, detalhadamente, os problemas encontrados no decorrer do projeto enunciado, como as dificuldades no estudo da máquina "QRC", nas ligações dos componentes e na maquinação do molde positivo. Além disso, são discutidas sugestões de melhorias e propostas para trabalhos futuros aplicáveis à máquina em questão.

Capítulo 2

Estado da arte

O fabrico de moldes positivos para próteses e ortóteses representa um grande desafio para as empresas desta área, que se encontra em constante evolução tecnológica, com o objetivo de aperfeiçoar os métodos de fabrico e obtenção tanto dos moldes positivos como dos dispositivos médicos.

Neste capítulo, realiza-se uma exposição e análise de temas importantes para a realização do presente projeto. São introduzidas informações sucintas sobre a área da ortoprotesia, incluindo definições essenciais para a compreensão das próteses, ortóteses e dos seus processos de fabrico, abordando as várias etapas dos mesmos. Ainda, é apresentada a diferença entre o fabrico tradicional e o fabrico digital destes dispositivos médicos, introduzindo a tecnologia CNC, que é o tema central deste projeto.

2.1 Introdução à Ortoprotesia

Quando se recorre a registos da história humana é possível perceber que todas as guerras e batalhas travadas ao longo do tempo foram determinantes para ocorrer um aumento do número de pessoas que precisam utilizar dispositivos médicos, como próteses ou ortóteses. Consequentemente, devido em grande parte à Primeira e Segunda Guerras Mundiais e, por isso, aos veteranos de guerra, a área da Ortoprotesia evoluiu, por haver uma grande necessidade em encontrar soluções para os problemas que surgiram da guerra.

O aumento do desenvolvimento de vários conceitos, nesta área, levou, igualmente, ao investimento e apoio financeiro, por parte de algumas entidades, para a pesquisa e desenvolvimento de novos *designs* e materiais para utilizar nestes dispositivos [2].

O termo **amputação** refere-se ao processo pelo qual uma parte do corpo é separada do mesmo, podendo ter diversas causas, sendo as mais comuns:

- Traumáticas: acidentes de viação ou de trabalho;
- Vasculares: doenças vasculares como a diabetes e doença arterial periférica;
- Infecções: infecções ósseas que podem levar à gangrena;
- Tumores: ósseos ou dos tecidos moles;
- Congénitas: malformações congénitas.

O tipo da amputação é descrito pela articulação ou parte óssea onde a mesma acontece [2], sendo os vários níveis de amputação mostrados pelas **Figuras 2.1a** e **2.1b**, conforme o membro afetado pela amputação.

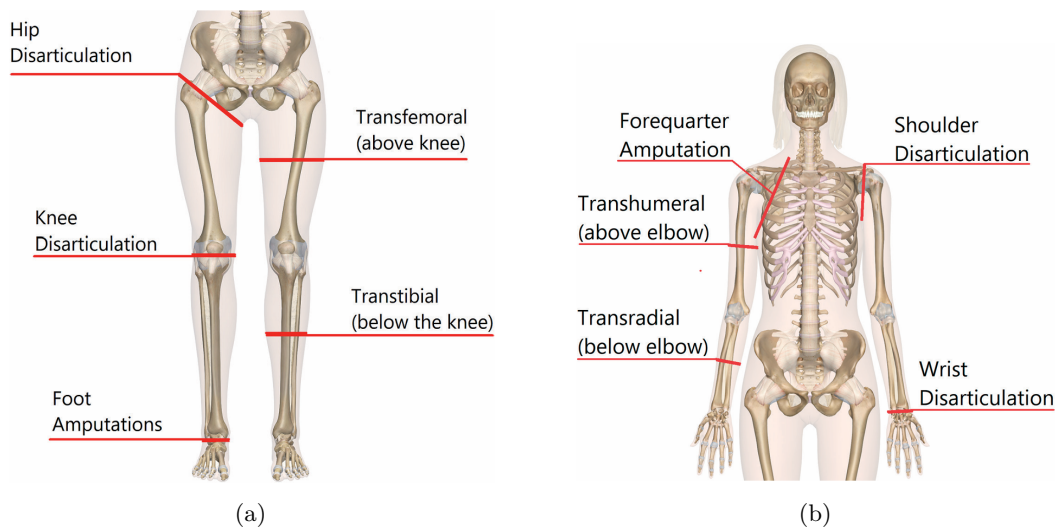


Figura 2.1: Amputações da extremidade inferior (a) e superior (b)
[3]

2.1.1 Próteses ou dispositivos protésicos

A **prótese**, ou dispositivo protésico, é um membro artificial, que tem a função de restaurar de modo funcional ou cosmético uma parte do corpo que se encontra em falta, e pode prolongar-se proximalmente com o intuito de suportar ou controlar a articulação anatómica adjacente [2] [4]. A parte do membro amputado remanescente, após a amputação, designa-se de **membro residual** e a parte da prótese que estará em contacto com o membro residual denomina-se **encaixe** protésico [2].

Ao enumerar os tipos de próteses existentes, é possível destacarem-se dois grandes grupos:

- **Próteses para amputações do membro inferior**, que têm o intuito de restaurar uma ou mais partes dos membros da parte inferior do corpo (pé, perna, coxa) que tenham sido amputados. Existe uma enorme variedade de componentes que podem ser aplicados a este tipo de próteses, nomeadamente, pés, articulações do tornozelo, joelhos, encaixes e diversos tipos de suspensão de prótese, selecionados conforme o nível de atividade e condição física do utilizador [4]. Através da **Figura 2.2** podem ser visíveis exemplos destes tipos de próteses;

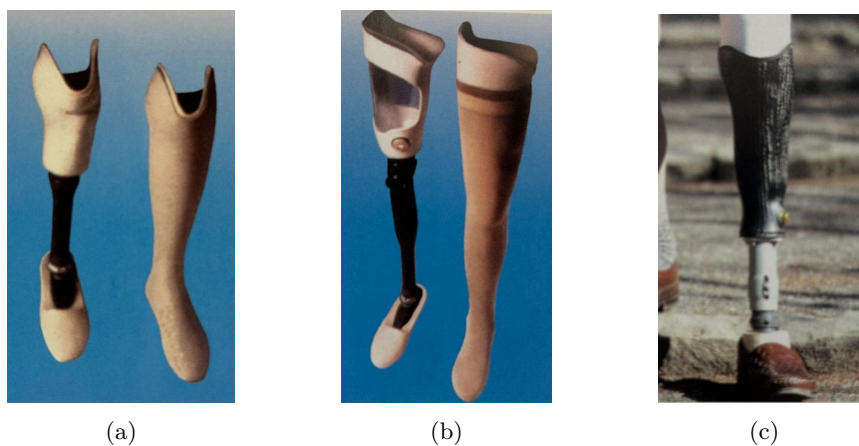


Figura 2.2: Exemplos de próteses desenhadas para amputações do membro inferior: transtibial (a) (c) e transfemural (b) [5]

- **Próteses para amputações do membro superior**, que têm o objetivo de restaurar uma ou mais partes dos membros da parte superior do corpo (mão, antebraço, braço) que tenham sido amputados [4]. Tal como nas próteses para a extremidade inferior, é possível a colocação de diversos componentes para a construção de uma prótese da extremidade superior, como mão, punho, cotovelo e ombro, sendo esta escolha efetuada segundo a funcionalidade que o utente pretende obter com a prótese. Na **Figura 2.3** podem ser observados alguns exemplos destas próteses.

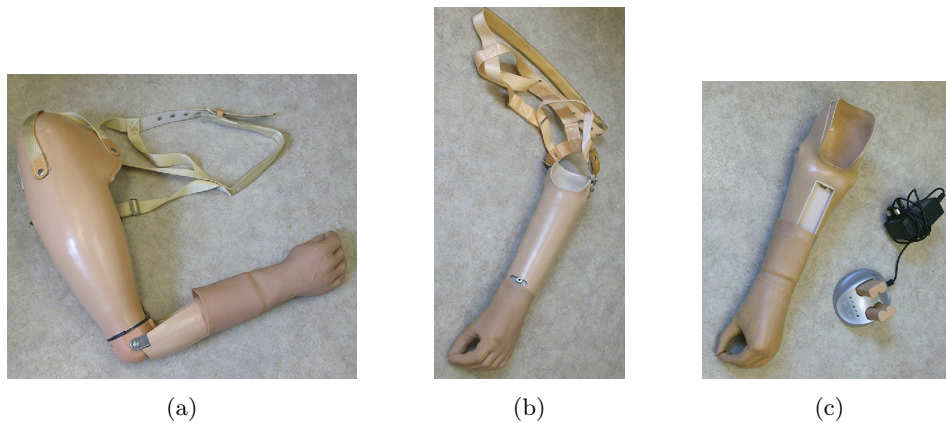


Figura 2.3: Exemplos de próteses desenhadas para amputações do membro superior: transradial (b) (c) e transumeral (a) [5]

Dentro dos dois grupos enumerados, existe uma grande variedade de próteses, conforme o nível da amputação, podendo as mesmas ser consultadas nas **Tabelas 2.1a e 2.1b**.

Tabela 2.1: Classificação do tipo de amputação e próteses do membro inferior (a) e do membro superior (b) [3] [4]

(a)

Nível amputação		
Próteses do membro inferior	<i>Hemipelvectomy</i>	Hemipelvectomy
	<i>Hip Disarticulation (HD)</i>	Desarticulação da anca
	<i>Above Knee (AK)</i>	Acima do joelho / Transfemural
	<i>Knee Disarticulation (KD)</i>	Desarticulação do joelho
	<i>Below Knee (BK)</i>	Abaixo do joelho / Transtibial
	<i>Ankle Disarticulation (AD)</i>	Desarticulação do tornozelo
	<i>Foot amputation</i>	Amputações do pé

(b)

Nível amputação		
Próteses do membro superior	<i>Forequarter</i>	Interescapulotorácica
	<i>Shoulder Disarticulation (SD)</i>	Desarticulação do ombro
	<i>Above Elbow (AE)</i>	Acima do cotovelo / Transumeral
	<i>Elbow Disarticulation (ED)</i>	Desarticulação do cotovelo
	<i>Below Elbow (BE)</i>	Abaixo do cotovelo / Transradial
	<i>Wrist Disarticulation (WD)</i>	Desarticulação do punho
	<i>Hand amputation</i>	Amputações da mão

Para as próteses serem suportadas nos membros, são utilizadas estratégias de suspensão recorrendo a proeminências ósseas, sucção ou tirantes/cintos de suspensão. Como referido, é sempre necessário recorrer-se à utilização de um encaixe, para a colocação do membro residual, que pode ter ou não uma interface, entre o membro

e o próprio encaixe, para torná-lo mais confortável. Ainda, existem diversos componentes que podem ser aplicados às próteses, consoante o seu tipo, os critérios e objetivos que se pretendem com a mesma. Deste modo, aquando da sua seleção é importante considerar alguns critérios tais como a idade, condição do membro amputado, nível de atividade e atividades que o utente pretende realizar com a prótese [4].

Com os cuidados adequados e, com o trabalho de uma equipa de saúde multidisciplinar, o amputado, indivíduo que sofreu a amputação, pode voltar às suas atividades de vida diária e encontrar-se apto para realizar qualquer tipo de trabalho ou ocupação que pretenda [2].

2.1.2 Ortóteses ou dispositivos ortésicos

Uma **ortótese** é um dispositivo externo utilizado na reabilitação de utentes com problemas neuromusculares e musculoesqueléticos, sendo, por isso, aplicado ao corpo humano com o intuito de restringir ou melhorar o movimento e, até mesmo, imobilizar alguma articulação, dando suporte a um segmento corporal [6].

Normalmente, a ortótese é definida segundo as articulações anatómicas abrangidas pela mesma e não pelo segmento corporal e, por isso, quando são mencionados os tipos de ortóteses existentes, é relevante falar em três grandes grupos [4]:

- **Ortóteses para o membro inferior**, que proporcionam uma melhoria da mobilidade ou imobilização da parte inferior do corpo (pé, perna, coxa). Para este tipo de ortóteses, tal como nas próteses, existe uma ampla gama de componentes para auxiliar as articulações afetadas, como articulações do tornozelo, do joelho e, até mesmo da anca [4]. Pela **Figura 2.4** conseguem perceber-se alguns exemplos deste tipo de ortóteses;

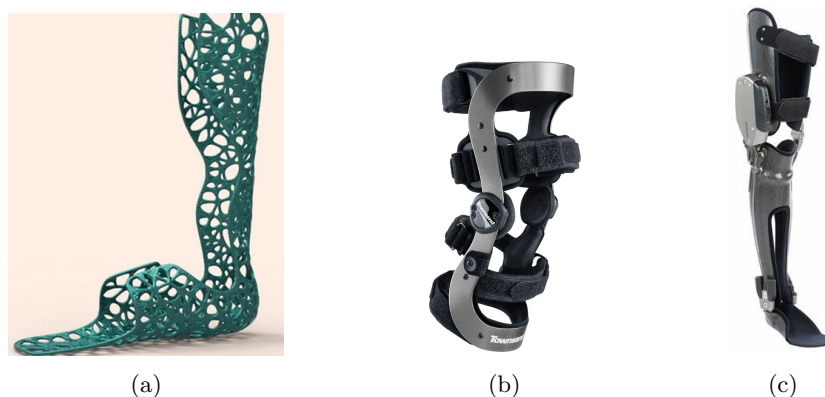


Figura 2.4: Exemplos de ortóteses desenhadas para o membro inferior: tornozelo-pé (a), joelho (b) e joelho-tornozelo-pé (c) [7] [8]

- **Ortóteses para o membro superior**, têm a finalidade de melhorar o grau de movimento ou imobilizar a parte superior do corpo (mão, punho, cotovelo). Nestas ortóteses encontram-se, igualmente, disponíveis diversos componentes para aplicar nas mesmas, como articulações do cotovelo e ombro [4]. Na **Figura 2.5** observam-se alguns exemplos deste tipo de ortóteses;



Figura 2.5: Exemplos de ortóteses desenhadas para o membro superior: mão-punho (a) (b) (c) [7] [9]

- **Ortóteses para a coluna**, utilizadas para corrigir deformidades da coluna ou evitar a progressão de desvios na mesma [4]. Diferente das ortóteses mencionadas acima, neste tipo de ortóteses normalmente não existem componentes externos, como articulações, que podem ser incluídos na estrutura. A sua conceção passa então pela escolha do melhor tipo de termoplástico, conforme o objetivo pretendido com a ortótese. É possível serem observados exemplos destas ortóteses através da **Figura 2.6**.

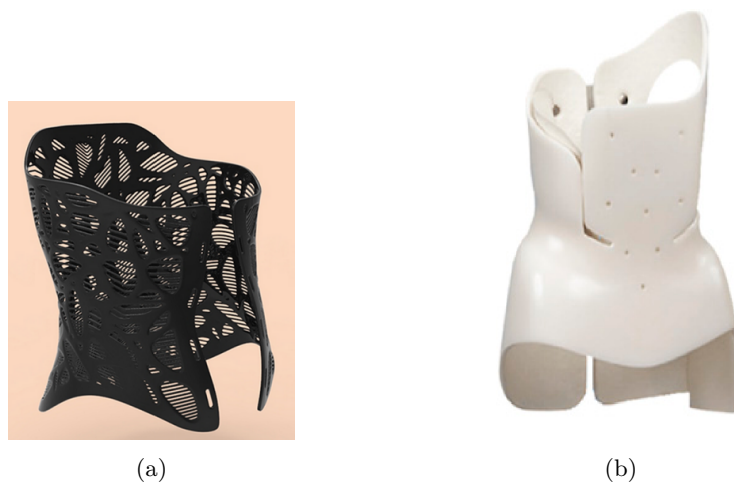


Figura 2.6: Exemplos de ortóteses desenhadas para a coluna (a) (b) [7] [10]

Tal como nas próteses, existe uma grande variedade de ortóteses, conforme a abrangência da articulação que a mesma implica. Através das **Tabelas 2.2, 2.3 e 2.4** podem ser verificadas as diferentes ortóteses que podem ser desenvolvidas [4].

Tabela 2.2: Classificação do tipo de ortótese para o membro inferior [3] [4]

Tipo ortótese		
Ortóteses do membro inferior	<i>Hip Knee Ankle Foot Orthosis (HKAFO)</i>	Ortótese anca Joelho-tornozelo-pé
	<i>Hip Orthosis (HpO)</i>	Ortótese da anca
	<i>Knee Ankle Foot Orthosis (KAFO)</i>	Ortótese joelho-tornozelo-pé
	<i>Knee Orthosis (KO)</i>	Ortótese joelho
	<i>Ankle Foot Orthosis (AFO)</i>	Ortótese tornozelo-pé
	<i>Foot Orthosis (FO)</i>	Ortótese pé

Tabela 2.3: Classificação do tipo de ortótese para o membro superior [3] [4]

Tipo ortótese		
Ortóteses do membro superior	<i>Shoulder Elbow Wrist Orthosis (SEWO)</i>	Ortótese ombro-cotovelo-punho
	<i>Shoulder Orthosis (SO)</i>	Ortótese ombro
	<i>Elbow Wrist Hand Orthosis (EWHO)</i>	Ortótese cotovelo-punho-mão
	<i>Elbow Orthosis (EO)</i>	Ortótese cotovelo
	<i>Wrist Hand Orthosis (WHO)</i>	Ortótese punho-mão
	<i>Wrist Orthosis (WO)</i>	Ortótese punho
	<i>Hand Orthosis (HO)</i>	Ortótese mão

Tabela 2.4: Classificação do tipo de ortótese para a coluna [3] [4]

Tipo ortótese		
Ortóteses coluna	<i>Cervical-thoracic-lumbosacral Orthosis (CTLSO)</i>	Ortótese coluna
	<i>Cervical Orthosis (CO)</i>	Ortótese cervical
	<i>Toracic-lumbosacral Orthosis (TLSO)</i>	Ortótese torácica-lombo-sacral
	<i>Toracic Orthosis (TO)</i>	Ortótese torácica
	<i>Lumbosacral Orthosis (LSO)</i>	Ortótese lombo-sacral
	<i>Lumbar Orthosis (LO)</i>	Ortótese lombar
	<i>Sacroiliac Orthosis (SIO)</i>	Ortótese sacro-ilíaca

A decisão pelo melhor tipo de ortótese, conforme a disfunção ou limitação funcional do segmento corporal, passa pela compreensão de duas questões importantes, por parte dos profissionais de reabilitação: 1) princípios básicos de biomecânica do movimento da articulação em referência e, 2) o *design* subjacente da ortótese, bem como do seu impacto na articulação em causa [6]. Também é necessário considerar o tipo de material, ou articulação a incluir de modo a ser encontrado um compromisso

entre força e função máxima necessária, com o menor peso possível do dispositivo e menor envolvimento da articulação envolvida [4].

Tal como no processo de fabrico de próteses, para ser construída o melhor tipo de ortótese para cada utente é necessário recolher-se uma variedade de informação, tais como medidas ou traçados do membro, ou segmento corporal, que irá utilizar a ortótese. Seguidamente, no **Subcapítulo 2.2** é dado ênfase a este processo de toma de medidas e moldes, uma vez que é de extrema relevância para o projeto em causa.

2.2 Fabrico de próteses e ortóteses

Como referido, anteriormente, a área da Ortoprotesia apresenta uma longa e rica história no que diz respeito à pesquisa e desenvolvimento de dispositivos ortopédicos para restaurarem a função e, até mesmo, melhorarem certas condições médicas. O intuito destes dispositivos é tornarem-se cada vez mais inovadores a nível anatómico e tecnológico, para responder da melhor forma às consequências clínicas que advêm de situações como as amputações dos membros ou deformidades decorrentes de algumas patologias.

É certo que este progresso tecnológico se encontra referido ao longo da história da humanidade, no entanto, apenas se tornou significativo no século XX, como consequência das guerras mundiais, ao haver um aumento da procura destes dispositivos médicos [11].

Um dos maiores avanços a nível tecnológico, nesta área, foi a introdução de recursos digitais, como é o caso de *scanners* 3D, para a toma de moldes necessários para o fabrico de próteses e ortóteses, ou, ainda, a utilização de máquinas CNC no fabrico dos moldes positivos, para o mesmo objetivo. Como mencionado, anteriormente, no **Capítulo 1**, tudo isto permite a redução de tempo, recursos e custos no fabrico destes dispositivos médicos.

Para que uma prótese ou ortótese seja aplicada a um utente, é essencial serem seguidos uma série de passos importantes até ser obtido o produto final - dispositivo médico realizado à medida. Assim, habitualmente, são seguidos seis passos importantes listados na **Tabela 2.5**, sendo que, para a presente dissertação, apenas serão abordados os passos 2 a 4. Pela **Tabela 2.5** é possível perceber que, após ser definida a prescrição médica para o tipo de prótese ou ortótese a utilizar pelo utente, seguem-se uma série de passos para ser fabricado o melhor dispositivo para o mesmo.

Tabela 2.5: Passos essenciais para a aplicação de dispositivos médicos [2]

Passos para aplicação de dispositivos médicos	
1.	Avaliação/prescrição
2.	Tiragem medidas/molde
3.	Fabrico do dispositivo/alinhamento
4.	Aplicação do dispositivo/alinhamento estático
5.	Modificação do dispositivo/alinhamento dinâmico
6.	Reavaliação/follow up

Em seguida, são apresentados os processos, tradicional e digital, para a obtenção dos moldes positivos, necessários para o fabrico de próteses e ortóteses.

2.2.1 Processo tradicional de fabrico de moldes positivos

Quando é referido o processo tradicional para o fabrico de moldes positivos, em Ortoprotesia, é importante mencionar que todos os passos realizados até à obtenção dos mesmos, exigem o recurso de materiais como ligaduras gessadas, visíveis na **Figura 2.7a**, ou de resina, como a **Figura 2.7b** mostra.



(a)



(b)

Figura 2.7: Ligaduras gessadas (a) [12] e de resina (b) [13]

Toma de medidas do membro ou segmento corporal

Para serem atingidos melhores resultados a nível dos moldes positivos dos membros ou segmentos corporais e, posteriormente, dos dispositivos médicos, é importante serem tiradas uma série de medidas, muitas vezes referenciadas por pontos anatómicos, ósseos. Assim, algumas destas medidas antropométricas, consideradas importantes, são comprimentos dos membros ou membros residuais, circunferências e, ainda, dimensões ântero-posterior (AP) e médio-lateral (ML). Todas as medidas tiradas são, devidamente, registadas em formulários específicos, como se pode ver pela **Figura 2.8**, consoante o segmento corporal em estudo, para posteriormente serem analisadas aquando da retificação do molde positivo e do fabrico do dispositivo médico [11].

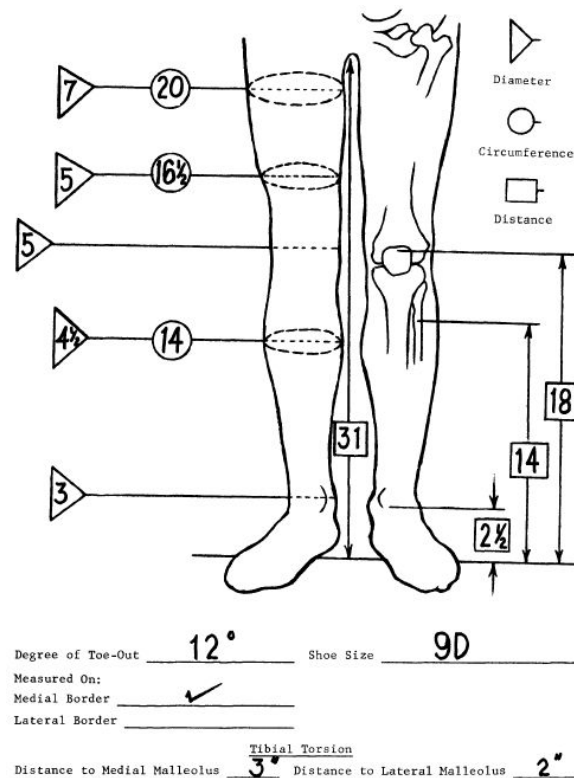


Figura 2.8: Exemplo de um formulário de medidas para o fabrico de uma órtese para o membro inferior [14]

Molde negativo e positivo (3D) do membro ou segmento corporal

Após serem tiradas todas as medidas relevantes para o processo de fabrico do dispositivo médico, é, então, necessário proceder-se ao fabrico do molde negativo, que corresponde a uma impressão negativa, tirada ao segmento do corpo que se pretende transpor para um molde positivo, ou modelo 3D, sem o qual é impossível fabricar a órtese ou prótese. Assim, como referido no **Subcapítulo 2.2.1**, são utilizadas ligaduras gessadas ou de resina para este efeito. Nesta fase, é necessário realizarem-se alguns procedimentos:

1. Colocar uma meia tubular ou película adesiva, sobre a pele, para proteger a mesma e para conter tecidos moles, que possam existir, no molde, como se observa na **Figura 2.9a**;
2. Marcar todas as proeminências ósseas e pontos anatómicos importantes, com tinta indelével, para os mesmos serem, posteriormente, observados no molde positivo;
3. Aplicar uma camada fina de ligadura gessada ou de resina sobre o membro, conforme a **Figura 2.9b** mostra, de modo a cobrir todas as partes relevantes.

Em algumas situações, nesta fase, são realizadas pressões no membro de modo a serem obtidos pontos relevantes do mesmo, no molde positivo;

4. Retirar o molde negativo do membro, de modo que a sua forma e contornos sejam mantidos, bem como o seu correto alinhamento, como se verifica na **Figura 2.9c**.

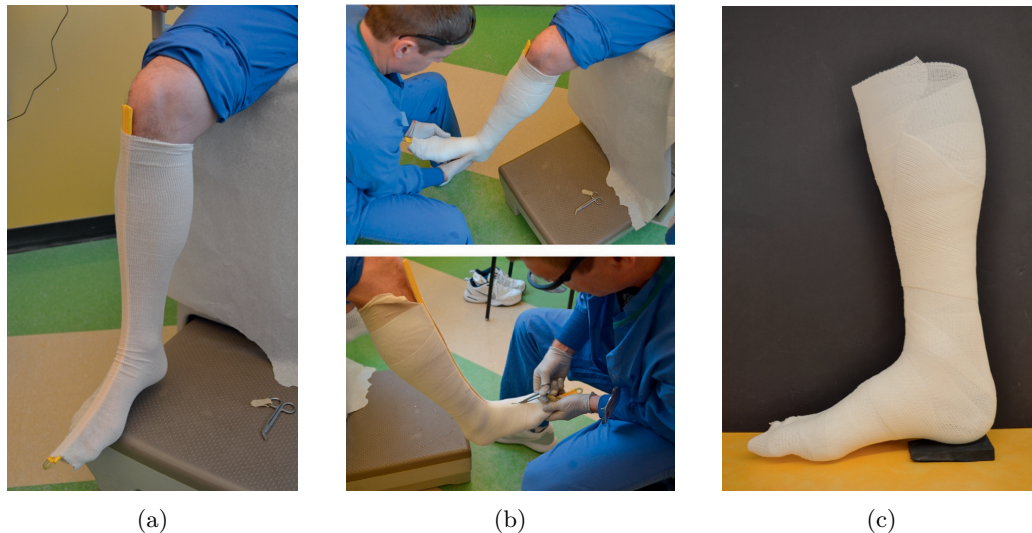


Figura 2.9: Procedimentos realizados aquando da toma do molde negativo, através do método tradicional [11]

Em casos específicos de alguns dispositivos médicos (próteses cosméticas), quando se quer maior detalhe e precisão, são utilizados outros tipos de materiais para a toma do molde, como o caso de silicone de grau médico ou, ainda, alginato, muitas vezes utilizado em medicina dentária.

Depois de ser obtido o molde negativo, procede-se ao fabrico do molde 3D positivo. Pelo método tradicional, é importante certificar que o molde negativo se encontra bem selado, para que, quando o mesmo for cheio com gesso, não ocorra nenhuma fuga de material. Para auxiliar na posterior retificação do molde positivo, aquando da colocação do gesso no molde negativo, é introduzido um tubo dentro deste, para ser possível prender o molde positivo a um torno.

Quando o molde se encontrar no estado sólido, é retirada a impressão negativa, verificando que todos os pontos marcados, anteriormente, na toma do molde, foram transpostos para o molde positivo [11].

Retificação do molde positivo de acordo com os parâmetros desejados

Para se obter o molde positivo correto, e mais próximo do resultado pretendido, é necessário realizarem-se uma série de procedimentos para a obtenção do mesmo, que serão enumerados de seguida. É importante referir que todas as modificações e retificações mencionadas realizam-se de acordo com o objetivo que se pretende obter com o dispositivo, bem como da experiência do técnico Ortoprotésico [11]:

1. Remover qualquer tipo de artefatos que possam ter ocorrido durante o processo de obtenção do molde negativo ou, até mesmo, do processo de enchimento do mesmo;
2. Realizar as alterações ou retificações, necessárias, conforme o inicialmente estabelecido para a produção do dispositivo médico. Nesta fase é importante confirmar, através do formulário de medidas preenchido anteriormente, se as medidas tiradas se encontram dentro do que é pretendido, no molde positivo, e, caso necessário, alterar o mesmo para as medidas corretas;
3. Adicionador gesso, nas proeminências ósseas, de modo a fornecer zonas de descompressão no dispositivo médico, porque apesar das modificações anteriores serem realizadas e o molde positivo representar a forma 3D do membro ou segmento corporal, não se conseguem obter, através deste, densidades dos tecidos;
4. Remover gesso para que deste modo sejam aplicadas pressões, toleradas, aquando da aplicação do dispositivo, processo que pode ser observado através da **Figura 2.10**.



(a)



(b)

Figura 2.10: Retificação do molde positivo, através do método tradicional [11]

Após terem sido realizadas as alterações pretendidas e necessárias, o molde positivo tem de ser deixado o mais liso possível, sem superfícies abrasivas, e encontra-se, assim, pronto para se proceder ao fabrico da prótese ou ortótese. Este processo é, abordado, à frente, no **Subcapítulo 2.2.3**.

2.2.2 Processo digital de fabrico de moldes positivos

Quando se fala em tecnologia e processos digitais, na área da Ortoprotesia, para a obtenção de moldes positivos, o mais comum é o recurso à tecnologia *Computer Aided Design/Computer Aided Manufacturing* (CAD/CAM). Esta tecnologia, apenas foi introduzida na década de 1960, e utilizava, inicialmente, estereografia e digitalização para criar um modelo numérico, guiando uma máquina fresadora à obtenção de um modelo 3D de um membro residual.

Hoje em dia, a maioria dos sistemas *Computer Aided Design* (CAD) utilizam dispositivos que registam informações digitais de um segmento corporal, como é o caso dos *scanners*, exemplificado pela **Figura 2.11**, captando, deste modo, os segmentos corporais desejados.



Figura 2.11: Exemplo de um *scanner* 3D, para obtenção de molde positivo [15]

Após ser realizada a captação da imagem 3D do segmento corporal, ocorre a tradução da mesma em ficheiros que são trabalhados em *softwares* CAD próprios, como é o caso do *software* exibido pela **Figura 2.12**.

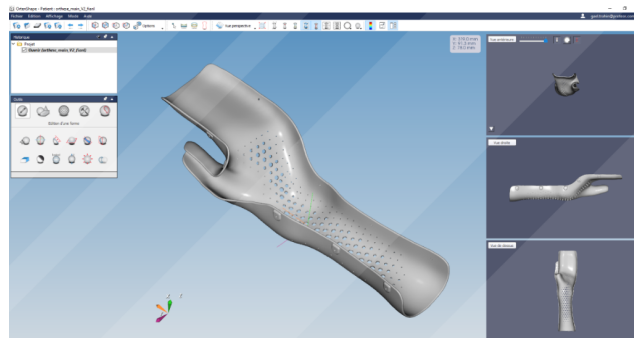


Figura 2.12: [16]

Quando o modelo 3D se encontra definido é enviado para um sistema CAM, como, por exemplo, uma máquina de fresar CNC, que a **Figura 2.13a** mostra, obtendo-se, assim, o molde positivo em PU, do segmento corporal digitalizado, visível na **Figura 2.13b** [11].

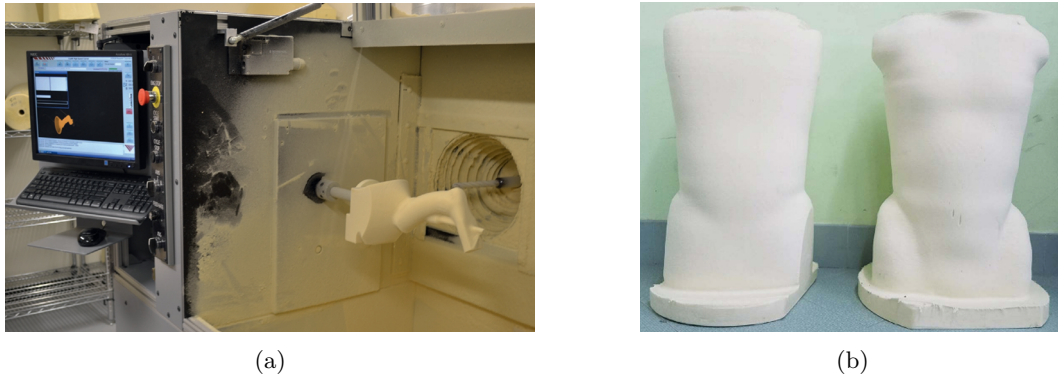


Figura 2.13: Exemplo de uma máquina de fresar CNC (a) e a obtenção de um molde positivo (b) através da mesma [11]

Em suma, no processo digital, os componentes ou materiais necessários para o efeito consistem num dispositivo de digitalização, habitualmente um *scanner* 3D, um computador pessoal (PC) e uma máquina de fresar CNC. A informação obtida pelo processo de digitalização é processada pelo PC, traduzida num ficheiro com pontos 3D e criado um modelo gráfico com o segmento corporal desejado. Por fim, todos os dados obtidos e modificados são enviados para uma máquina de fresar CNC com o intuito de ser obtido o molde positivo. Seguidamente, são explorados estes processos [11].

Toma de medidas do membro ou segmento corporal

O processo da toma de medidas do segmento corporal é, em tudo, idêntico, ao processo descrito no **Subcapítulo 2.2.1**.

Molde positivo tridimensional do membro ou segmento corporal

Tal como no processo tradicional de fabrico de moldes positivos, após serem tiradas todas as medidas relevantes para o processo de fabrico do dispositivo médico, é, então, necessário proceder-se ao fabrico do molde positivo. Nos processos digitais, consegue-se excluir uma das etapas do método tradicional - molde negativo - uma vez que através da digitalização do segmento corporal, como se pode ver na **Figura 2.14**, com o auxílio de um *scanner* 3D, idêntico ao que se pode observar, acima, na **Figura 2.11**, é possível obter-se, instantaneamente, um modelo 3D da parte do corpo desejada. Com este processo, é possível observar-se que existe uma economia de tempo na obtenção do molde positivo [11] [17].



Figura 2.14: [15]

Para este processo é necessário realizarem-se menos procedimentos, quando comparado com o método tradicional:

1. Marcar todas as proeminências ósseas e pontos anatómicos importantes, com tinta indelével, para os mesmos serem, posteriormente, observados no molde positivo (alguns *softwares* CAD permitem este aspeto);
2. Preparar o equipamento necessário para se proceder à digitalização do membro - PC e *scanner*, ou, em alguns casos, *tablet* ou *smartphone* com *scanner* próprio;
3. Pedir ao utente para manter uma posição o mais estático possível, para a digitalização ser a mais correta;
4. Digitalizar o segmento corporal desejado;
5. Verificar que foi gerada uma nuvem de pontos correspondente ao membro digitalizado. Caso este processo não vá ao encontro do objetivo pretendido, repetir os passos 4 e 5 [18].

De modo a ser obtido o melhor modelo 3D do segmento corporal, são vários os tipos de *scanner* que podem ser utilizados para este processo, podendo ou não ser portáteis. Normalmente, e com o intuito de se obter um erro menor, são utilizados *scanners* que têm por base o laser e tecnologia de luz estruturada, sendo que estes, normalmente, têm de ser ligados a um PC para poder ser visualizado o modelo 3D que se obtém com a digitalização.

Outra opção, mais versátil, no que diz respeito aos *scanners*, são os *scanners* portáteis de mão, onde se pode, igualmente, obter uma boa qualidade da digitalização, com a vantagem de poderem ser transportados mais facilmente. Habitualmente, estes tipos de *scanner* podem ser acoplados a, por exemplo, *tablets* ou *smartphone*, facilitando o processo de digitalização [18].

Retificação do molde positivo de acordo com parâmetros desejados

Ao ser utilizado o processo digital para a obtenção do molde positivo, é necessário recorrer-se a um *software* adequado para o modelo 3D ser alterado e, deste modo, ser obtido o molde positivo adequado para o processo de fabrico do dispositivo médico.

Com as ferramentas adequadas presentes no *software* CAD, escolhido pelo Ortoprotésico, são realizadas todas as modificações significativas, idênticas às realizadas na fase de retificação do molde positivo do processo tradicional, referidas no **Subcapítulo 2.2.1**. A grande vantagem da utilização de um processo digital é poder ver, em tempo real, as modificações realizadas ou forças aplicadas no molde. Estas alterações dependem, na maioria das vezes, da manipulação correta do modelo 3D, pelo que, atualmente, podem ser encontradas algumas opções, dentro do *software*, que simplificam este processo, como é o caso de *templates* com pontos anatómicos ou com áreas a aplicar pressão, como é possível observar na **Figura 2.15**.

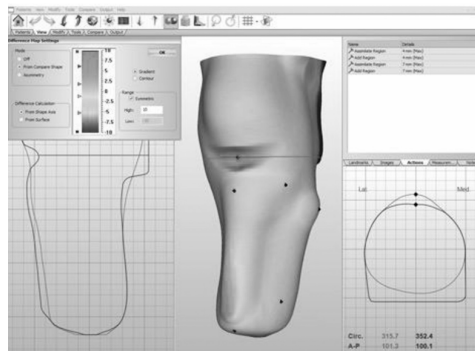


Figura 2.15: [19]

Outro aspeto positivo, é a capacidade do modelo 3D do segmento corporal permitir que sejam incluídos os aspetos biomecanicamente definidos pelo técnico, aquando da realização da prótese ou ortótese. Quer isto dizer que podem ser desenhadas, através do *software*, as linhas de corte dos dispositivos, exibidas pela **Figura 2.16**, bem como adicionadas algumas "peças" que poderão auxiliar o fabrico do mesmo dispositivo. A grande vantagem é que estas opções podem ser observadas nos vários planos anatómicos - anterior, posterior, lateral e medial - do molde positivo, ao mesmo tempo [11] [18].

Existe uma grande variedade de *softwares* CAD/CAM disponível para a retificação, *design* e fabrico de próteses e ortóteses, consoante o resultado final que se pretende obter, este aspeto é abordado à frente, no **Capítulo 3** [18].

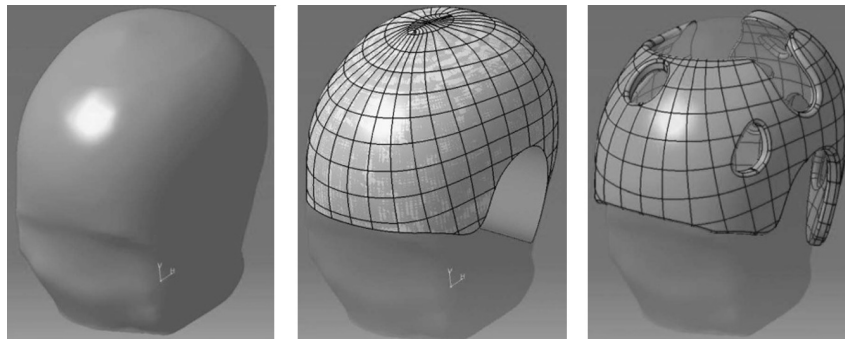


Figura 2.16: Processos possíveis através do *software* de modelação [20]

Produção do molde positivo de acordo com os parâmetros desejados

Após ser obtido o molde ou modelo 3D desejado a partir da digitalização do segmento corporal, é necessário converter o mesmo num ficheiro .stl ou .step, tendo em consideração que a máquina de fresar CNC permita. O ficheiro é então convertido, com o auxílio de um *software* CAM, normalmente, disponibilizado pelo fabricante da máquina fresadora, para assim poder ser utilizado na máquina e ser obtido um molde 3D em PU [18] [11].

Quando o modelo 3D digital se encontra pronto, a máquina de fresagem é preparada para ser obtido o molde positivo desejado. Como cada tipo de dispositivo médico é diferente e personalizado, normalmente são necessários parâmetros distintos de fresagem, conforme o que se pretende obter. Assim, por exemplo, quando se pretende obter formas mais cilíndricas e ovais, habitualmente encontradas nos encaixes para as próteses, recorre-se à fresagem do tipo torno, por outro lado, quando se fala em ortóteses, mais propriamente fresagem da parte dos pés, o *setup* de fresagem é, normalmente, vertical, pois esta estrutura assemelha-se a uma placa e assim conseguem-se obter outro tipo de acabamentos [11].

2.2.3 Fabrico da ortótese ou prótese através do molde positivo

Após ser obtido o molde positivo, pela via "tradicional" ou pela via digital, é necessário proceder-se ao fabrico da prótese ou ortótese. O processo de fabricação a partir do molde positivo envolve várias etapas críticas que garantem a precisão, funcionalidade e conforto do dispositivo final. Em seguida, são apresentados alguns dos processos efetuados até à obtenção do dispositivo médico:

1. Preparação do molde positivo: como referido anteriormente, é esperado que o molde positivo represente uma réplica exata, após certo tipo de modificações, do segmento corporal que se pretende, assim o molde deve ser verificado para garantir que as dimensões se encontram corretas. Apesar de no processo de retificação se pretender que o molde se encontre suave e sem imperfeições

superficiais, algumas vezes é necessário verificar que o mesmo se encontra neste estado. É, ainda, necessário isolar o molde, antes de ser aplicado o material com que irá ser fabricado o dispositivo, para evitar que o mesmo adira ao mesmo. Assim, são aplicados agentes de libertação como "sacos" de álcool polivinílico (PVA) para o efeito [6].

2. Seleção dos materiais para o fabrico do dispositivo: é uma das etapas-chave do processo, pois determinará a flexibilidade, resistência, durabilidade e peso do mesmo. Para o fabrico são selecionados materiais como termoplásticos, de diversas durezas, resinas acrílicas, epóxi ou de PU e fibras de carbono e vidro. É compreensível que, por exemplo, as próteses para a extremidade inferior, por suportarem o peso corporal do utilizador, necessitem de materiais mais robustos e com maior durabilidade, e, por outro lado, as ortóteses para a coluna ou extremidade superior utilizem materiais mais leves e flexíveis [6].
3. Processo de moldagem: após a preparação do molde positivo, explicado acima, e a seleção dos materiais a utilizar, segue-se o processo, propriamente dito, do fabrico do dispositivo, denominado moldagem. É possível obter-se, de acordo com o dispositivo a fabricar, por processos de moldagem diferentes:
 - *Moldagem por compressão*: utiliza pressão para conformar o material ao molde positivo, sendo que é o processo mais utilizado quando se aplicam materiais termoplásticos, como o polipropileno (PP) ou polietileno (PE). Este processo pode ser visualizado na **Figura 2.17**;



(a)



(b)

Figura 2.17: Termomoldagem de uma ortótese para o membro inferior [11]

- *Moldagem por laminação*: consiste na aplicação de camadas de material sobre o molde positivo e cada camada é impregnada com resina, como se observa na **Figura 2.18**. Normalmente, utiliza-se este processo para o fabrico dos encaixes para as próteses, aplicando-se, por exemplo, as fibras de vidro ou carbono. Após a aplicação da resina sobre as fibras escolhidas, o conjunto é curado para formar a estrutura final.

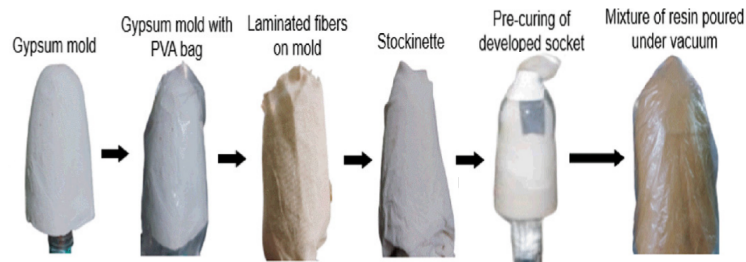


Figura 2.18: Moldagem por laminação de um encaixe de uma prótese para o membro inferior [21]

Em alternativa aos passos 1 a 3, enumerados acima, é possível, obter-se o dispositivo médico através do processo de fabrico aditivo, recorrendo a uma impressora 3D para este efeito, como se pode verificar na **Figura 2.19**. Caso se opte pela via digital para a obtenção do molde positivo, é necessário fazer o desenho do dispositivo, recorrendo a um *software* CAD, para assim ser obtido o ficheiro .stl correspondente ao dispositivo que se deseja fabricar [22]. Se o molde positivo for obtido pelo método tradicional, é necessário fazer-se a digitalização recorrendo a um *scanner* 3D do molde positivo e realizar-se, posteriormente, o processo anterior descrito.

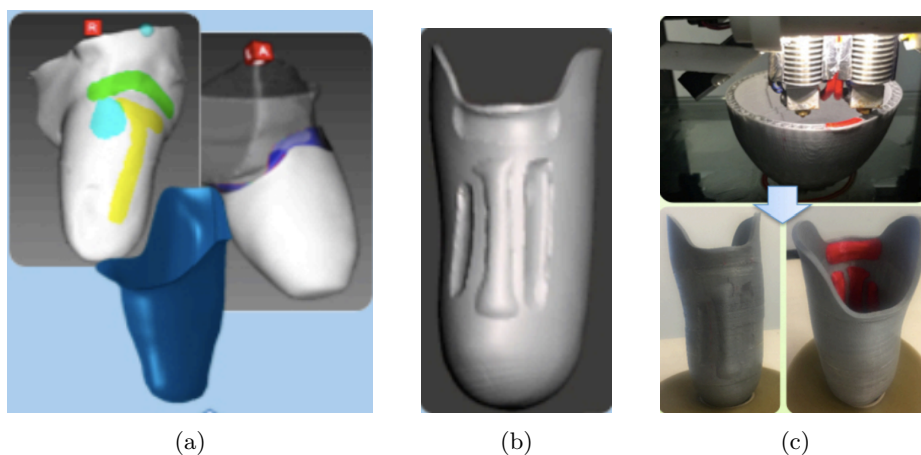


Figura 2.19: Obtenção de um encaixe de prótese para o membro inferior, recorrendo ao fabrico aditivo [22]

4. Ajustes e acabamentos: depois da moldagem ser realizada é necessário ajustar e acabar a prótese ou ortótese, que passa por cortar o material em excesso, conforme os contornos desejados, lixar e suavizar os bordos que ficam em contacto com a pele do utente. Além disto, é necessário acoplar componentes, que no caso das ortóteses são as tiras de fixação e as articulações externas, visíveis na **Figura 2.20a** e no caso das próteses são os componentes protésicos como pé, joelho, entre outros, como se pode ver na **Figura 2.20b**. Ainda, em alguns casos são adicionados forros no interior do dispositivo para aumentar o seu conforto e funcionalidade [6].

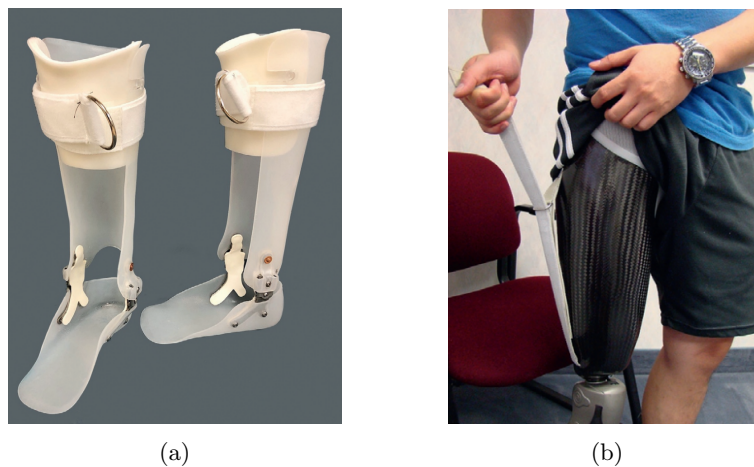


Figura 2.20: Exemplo de um par de ortóteses (a) e de uma prótese (b) para o membro inferior acabadas [6]

2.3 Conclusão do Capítulo 2

Neste capítulo, foram abordados os conceitos fundamentais relacionados com a área da ortoprotesia, incluindo termos técnicos referentes às próteses e ortóteses, para uma melhor compreensão deste ramo da saúde e para enfatizar a importância da personalização destes dispositivos, segundo as exigências dos seus utilizadores.

Foram, igualmente, explorados os avanços tecnológicos experienciados na área, apresentando uma evolução histórica das próteses e ortóteses, com foco nos seus métodos de obtenção. Analisaram-se os métodos tradicionais de fabrico e a introdução do fabrico digital, método mais recente que recebe destaque ao longo deste documento, incluindo o uso de ferramentas CAD/CAM e a utilização de máquinas CNC.

Por fim, foram detalhados os processos de fabricação de ortóteses e próteses, a partir de moldes positivos, desde a preparação do molde até à seleção dos materiais e métodos de moldagem. Estes processos são cruciais para garantir que estes dispositivos médicos atendam às necessidades específicas dos seus utilizadores.

Capítulo 3

Tecnologia CNC e maquinagem de materiais

O presente capítulo aborda a tecnologia CNC, essencial para a compreensão de diversos aspetos discutidos ao longo deste documento. Assim, é realizada uma análise detalhada desta tecnologia, abrangendo os tipos de máquinas CNC, os seus componentes, e os conceitos fundamentais, com referência ao que já está existente disponível no mercado. Além disso, é discutido o tema da maquinagem de diferentes materiais, com ênfase no PU, material utilizado neste projeto. Por fim, é destacada a importância da utilização do CNC e a sua aplicação na área da ortoprotesia.

3.1 Introdução ao *Computer Numeric Control* (CNC)

O *Numeric Control* (NC) consiste num tipo de tecnologia de automação, que tem por base uma série de comandos codificados através de códigos alfanuméricos e outros símbolos pré-definidos, com o intuito de tornar as operações das máquinas-ferramenta, como tornos, fresadoras ou retificadoras, possível. Estas operações são realizadas com o recurso a coordenadas cartesianas que definem a localização, direção e velocidade de movimentação da ferramenta de trabalho, utilizando, para isso, linguagens de programação específicas que desenvolvem os comandos necessários para este efeito. Através deste tipo de controlo, é possível automatizar as máquinas e ferramentas, permitindo uma maior precisão e eficiência no fabrico dos produtos.

Os comandos utilizados para controlar as máquinas-ferramenta são desenvolvidos através de *softwares* que descrevem as operações que as mesmas necessitam realizar para produzir o produto final. Na **Figura 3.1** é possível observar os vários passos deste processo [23].

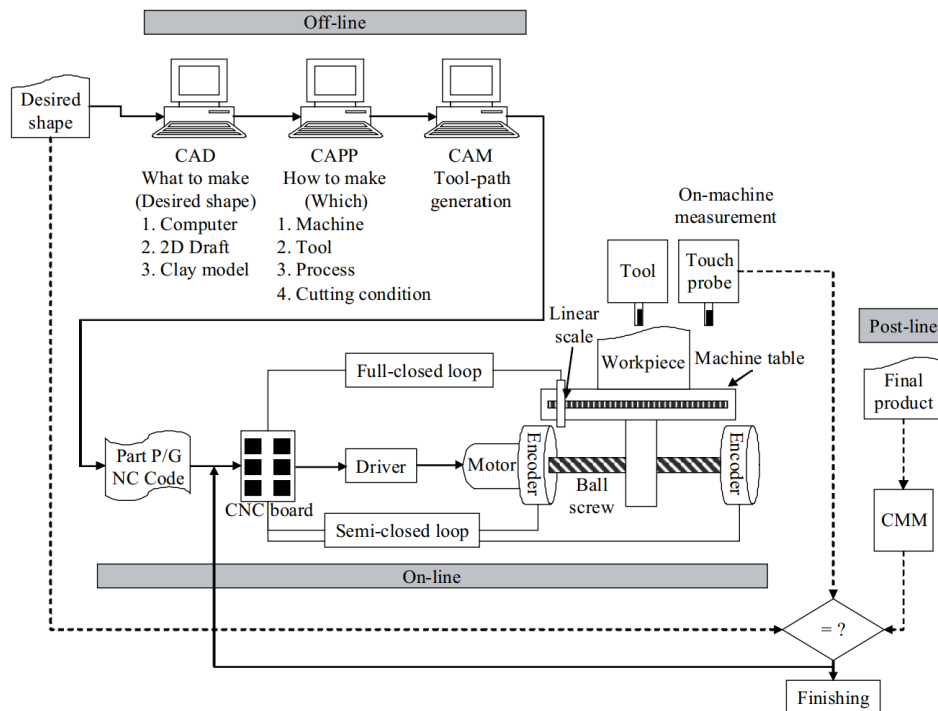


Figura 3.1: Arquitetura de uma máquina-ferramenta NC e respetivo fluxo de trabalho [23]

As **máquinas-ferramenta**, consistem em equipamentos de maquinagem, que permitem a realização de uma variedade de trabalhos como o corte, perfuração, pressão, retificação, entre outros, de modo rápido e preciso. São utilizadas, habitualmente, para o fabrico de determinados produtos, com especificações e precisões pré-determinadas. Após uma série de evoluções tecnológicas, estas máquinas são consideradas, hoje em dia, máquinas NC, pois evoluíram no sentido da precisão e aumento de velocidade de trabalho.

A utilização do NC teve início da década de 1940, quando foram desenvolvidas as primeiras máquinas-ferramenta controladas numericamente, sendo que o NC era utilizado através de dispositivos mecânicos e elétricos e evoluiu para a utilização de PCs e *software* próprio, para o efeito. Esta evolução foi fundamental para a introdução da tecnologia CNC e na indústria moderna.

O conceito de máquina-ferramenta, operada manualmente, foi introduzido após o desenvolvimento do primeiro motor a vapor, no fim do século XVIII. Após isto, foi inventado um método de controlo automático, através da tecelagem de tecidos recorrendo a uma máquina de tear que utilizava cartões perfurados, e foi aqui que

teve início o conceito de NC, apesar de só ter sido verdadeiramente aplicado depois da Segunda Guerra Mundial, quando a Força Aérea dos Estados Unidos da América (EUA) e a Parsons desenvolveram, em conjunto, uma maneira de mover dois eixos através da utilização de cartões perfurados com dados de coordenadas que permitiam maquinar peças de aeronaves. A partir desta altura, em 1952, o Massachusetts Institute of Technology (MIT), desenvolveu a primeira máquina-ferramenta de NC, uma fresadora com três eixos, que utilizava um tubo de vácuo, sendo o sistema NC maior que a própria máquina-ferramenta, devido à inexistência de placas de circuitos (transístores e circuitos integrados) para o seu funcionamento.

Ao analisar a evolução tecnológica do NC, verifica-se que a mesma evoluiu a par com os seus componentes. Como referido acima, inicialmente utilizavam-se circuitos de divisão de pulsos, operados pelo computador, recorrendo a tubos de vácuo e, além disso, a máquina-ferramenta era controlada por um motor de pressão a óleo, que controlava um relé, conforme o processamento lógico recebido. Por outro lado, com o aparecimento dos semicondutores e da sua aplicação ao NC, na década de 1960, a apresentação de motores elétricos e elementos de potência, na década de 1970 e, ainda, o desenvolvimento de componentes para computadores, na década de 1980, o NC recorrendo a *hardware* transitou para uma máquina NC com recurso a *software*, baseado nos *microcontroller unit* (MCU), tecnologia de potência elétrica e eletrónica e, ainda, tecnologia de *software*.

O intuito das máquinas NC foi o desenvolvimento de peças com formas complexas e com elevada precisão, pelo que é possível perceber que o NC foi, inicialmente, aplicado a fresadoras e máquinas de mandrilamento, contudo, com a necessidade de aumentar a produtividade e precisão em outras áreas, estendeu-se também às máquinas de eletroerosão e de laser.

Com a necessidade de automatizar os processos, cada vez mais se aplicou o NC às máquinas-ferramenta, sendo que para este efeito as mesmas eram controladas por um servomotor, consoante a operação que se desejava realizar e o tipo de mecanismo a ser acionado pelo mesmo. Quer isto dizer que o NC representa um dispositivo controlado através de comandos, gerados previamente, que ativam o servomotor e, assim, maquinam a peça final desejada. A partir desta combinação surgiu o CNC, pela combinação deste tipo de controlo (NC) com a tecnologia computadorizada (CNC).

O CNC é um sistema NC onde é utilizado um microcomputador como unidade de controlo de uma máquina-ferramenta. A grande diferença entre o NC e o CNC é que no primeiro a base de funcionamento são fitas ou cartões perfurados, que após a sua leitura transmitem a operação que se deseja realizar, tendo uma capacidade limitada para editar e automatizar o programa; por outro lado, o segundo, recorre a microcomputadores que processam e executam comandos codificados, definindo

as operações de funcionamento, permitindo, assim, uma maior flexibilidade e capacidade de controlo, por ser possível editar, facilmente, os programas, em tempo real, e integrá-los, com a ajuda de *softwares* CAD/CAM. Percebe-se, então, que as máquinas CNC são uma evolução das máquinas NC, pois utilizam computadores que processam e executam os comandos, para operar as máquinas-ferramenta [23].

3.1.1 Sistema de coordenadas nas máquinas CNC

Para se perceber o funcionamento de uma máquina CNC é extremamente importante perceber o sistema de coordenadas da mesma, sendo as posições relativas da ferramenta de corte relativamente à peça de trabalho definidas [24]. Para um objeto ser maquinado, a máquina tem de perceber qual a posição inicial da peça.

Habitualmente, as máquinas CNC utilizam um eixo de coordenadas X, Y e Z, para definir o espaço de coordenadas da mesma, no entanto, para se tornar mais fácil a sua programação, aplica-se um sistema onde a peça de trabalho se encontra estacionária e a ferramenta de corte se move relativamente à mesma. Assim, o sistema de coordenadas é definido relativamente à mesa de trabalho da máquina, como se pode analisar na **Figura 3.2** [25] [24].

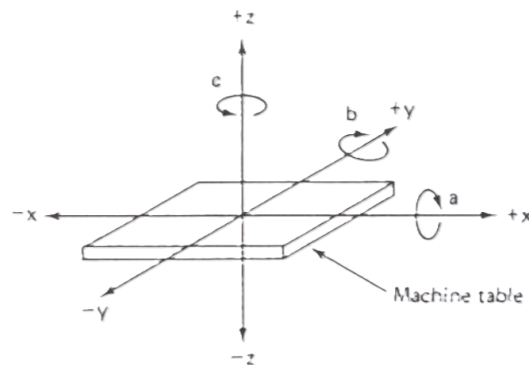


Figura 3.2: Exemplo de sistema de coordenadas de uma máquina CNC [24]

Pela **Figura 3.2** é possível verificar que os eixos X e Y, são definidos no plano da mesa de trabalho, enquanto o eixo do Z se encontra perpendicular à mesma, sendo o movimento na direção deste eixo, normalmente, controlado pelo movimento vertical do *spindle*. Também é possível perceber as direções de movimento positivas e negativas da ferramenta, relativamente à mesa, ao longo dos eixos X e Y [24]. Assim, o movimento ao longo do sistema de coordenadas baseia-se no movimento da ferramenta e não da mesa, ou seja, aumentando o valor da coordenada X, o movimento é para a esquerda, mas a ferramenta move-se para a direita ao longo da peça de trabalho. Por outro lado, quando a coordenada do eixo Z é aumentada, o *spindle* move-se para cima e quando é diminuída, move-se no sentido contrário, para baixo [25].

Ao serem combinados os eixos X, Y e Z, em ângulos de 90° , é possível ser criado um espaço 3D onde a máquina se consegue movimentar, sendo que cada eixo se encontra na origem. Para se compreender, mais facilmente, o sistema de coordenadas cartesianas destas máquinas, utiliza-se a **regra da mão direita**, como se observa na **Figura 3.3**, que consiste em apontar o terceiro dedo, da mão direita, para a ferramenta da máquina e verificar os outros dois primeiros dedos (polegar e indicador), quando formam um ângulo de 90° , entre eles. Assim, o terceiro dedo corresponde ao eixo Z, o indicador ao eixo Y e o polegar ao eixo X.

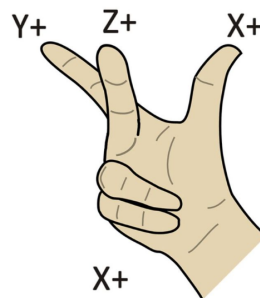


Figura 3.3: Regra da mão direita [25]

Quando a máquina é controlada ao longo de cada um dos seus eixos, cortando vários lados da peça no plano XY e com várias profundidades no eixo Z, consegue-se obter uma peça acabada através do material em bruto [25].

Uma máquina CNC configurada, necessita utilizar, corretamente, o sistema de coordenadas, estabelecendo uma origem. Ao produzir-se uma peça, a origem poderá ser qualquer ponto da mesma, mas deverá estar relacionada com as operações de maquinagem que serão realizadas. Como deste modo as coordenadas dos movimentos não coincidem com as da máquina, é necessário serem realizadas compensações, denominadas, *offsets* que determinam as coordenadas desejadas, calculando a diferença entre os valores da área de trabalho e a posição zero da máquina.

Normalmente, a linguagem mais utilizada para interpretar os comandos da máquina designa-se *Geometric Code* (G-code), podendo ser analisados alguns desses comandos, de, por exemplo, movimento e funcionalidades através da **Tabela 3.1**. A linguagem apresentada nesta **Tabela**, é um conjunto de linguagens que possibilitam comandos de controlo do motor e interrupções de algumas funções da máquina, sendo que alguns comandos incluem velocidade do *spindle*, motores de deslocamento dos eixos e, até mesmo, orientações da ferramenta num dado incremental ou absoluto.

O objetivo do G-code é sistematizar um procedimento de raciocínio 3D, direcionando os vários componentes da máquina através de um "caminho", configurando-os para a operação desejada ser efetuada e, assim, implementa o comando. Este objetivo tem de ser uma sincronia entre a programação do código pelo operador e as funções que a máquina permite.

Tabela 3.1: Endereços de comando numérico em G-code [26]

Letra	Descrição
N	Número da linha do comando
G	Indica movimento da ferramenta
X, Y, Z	Posição da ferramenta em 3D em relação aos eixos principais
U, V, W	Posição da ferramenta em 3D em relação aos eixos secundários
F	Taxa de alimentação da máquina CNC
S	Número de rotações do <i>spindle</i>
T	Quantidade de ferramentas utilizadas
M	Instruções sobre funções auxiliares da máquina
I, J, K	Parâmetros da interpolação circular
R	Raio do arco
A, B, C	Deslocamento de rotação dos eixos x, y e z
D	Determina o ponto de compensação do diâmetro da ferramenta
P	Comandos código G para CNC saltar em atraso ou em tempo

3.2 Composição das máquinas-ferramenta CNC

Uma máquina-ferramenta é considerada "máquina mãe" por ser através dela que são fabricadas vários tipos de máquinas. Com a necessidade do aumento da precisão dos processos industriais, houve uma evolução dos sistemas destas máquinas, tornando as máquinas-ferramenta convencionais em máquinas-ferramenta operadas por NC.

A construção de uma máquina, bem como os seus componentes, tem de apresentar uma elevada robustez para que a maquinagem seja realizada de modo preciso e eficaz. É, igualmente, crucial que a máquina seja resistente a variações na estrutura, como vibrações e alterações de temperatura [27]. Para garantir que estes fatores são atingidos, é fundamental a máquina ter mecanismos de monitorização e controlo, como codificadores e sensores. Como na presente dissertação o intuito é estudar uma máquina-ferramenta fresadora, é importante serem apresentados os componentes constituintes deste tipo de máquina.

Independentemente da função que a máquina irá desempenhar, os sistemas e componentes que a constituem são idênticos, podendo variar em alguns aspetos conforme a função principal que se pretende atingir com a mesma. Assim, normalmente, os sistemas que constituem uma máquina-ferramenta são os seguintes [23]:

- Estrutura;
- Sistema de acionamento;
- Sistema de controlo e sensores;
- Sistema de transmissão e conversão de movimento;

- Sistema de monitorização;
- Dispositivos de fixação de ferramentas e da peça.

Em seguida, são enumerados, detalhadamente, os componentes mais relevantes das máquinas CNC.

3.2.1 Estrutura

A estrutura mecânica da máquina CNC é onde se encontram acoplados todos os elementos constituintes da mesma, permitindo, ao mesmo tempo, suportar e mover a peça de trabalho e a ferramenta de corte. Consiste na base, coluna e mesa de trabalho, cada uma com a sua própria função.

A base da máquina fornece suporte e estabilidade à mesma, sendo normalmente desenhada em ferro fundido ou noutro tipo de material pesado, para reduzir as vibrações provenientes da máquina. A coluna, é a parte que suporta o cabeçote e a ferramenta de corte, estando ligada à estrutura principal, proporcionando assim rigidez. Por fim, a mesa de trabalho é a zona da máquina onde a peça a maquinar é fixada, podendo ou não ter peças de fixação que facilitam a fixação da peça à mesa.

Na **Figura 3.4** é possível verificar-se a estrutura de uma máquina CNC, que além de fixar os vários componentes da máquina procura evitar que ocorram deformações dos componentes, aquando das forças estáticas e dinâmicas, fornecendo estabilidade, alinhamento e precisão das partes móveis e, ainda, fazendo com que as vibrações sejam amortecidas. É importante que estes aspetos sejam conseguidos para assegurar o correto funcionamento da máquina [1] [27] [28].

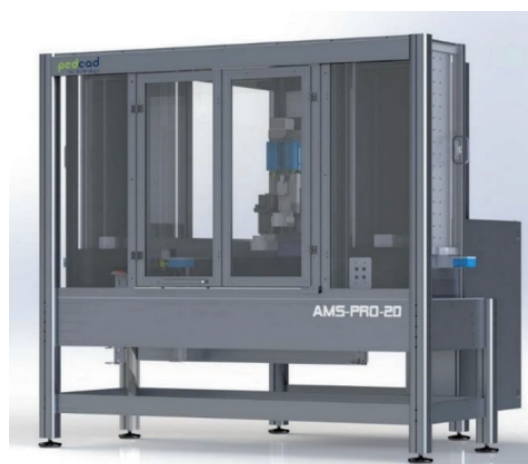


Figura 3.4: Exemplo de uma máquina de fresagem CNC [1]

3.2.2 Acionamento rotativo

Consiste nos sistemas que traduzem as ordens de comando em movimentos físicos da máquina, gerando a superfície de corte, executando a deslocação dos eixos e a rotação da árvore.

Quando se pretende obter um bom desempenho de uma máquina CNC, para atingir superfícies adequadas, através do processo de maquinagem, é importante haver coordenação de movimentos dos vários componentes da máquina: primários, que envolvem a alimentação dos eixos e da árvore para efetuar o corte; secundários, que preparam a maquinagem e garantem o sucesso da operação. Deste modo, os acionamentos podem ser rotativos ou lineares e devem garantir um movimento contínuo e preciso. A seguir são resumidos estes sistemas [27].

Acionamento dos eixos

O controlo, em malha fechada, do movimento dos eixos é realizado com recurso, a um servo sistema composto pelo controlador, conversor de potência, servomotor e respetiva unidade de medição. O controlador recebe o sinal de comando, fazendo o motor girar, que depois é enviado a um fuso, rodando-o. Esta rotação traduz-se em movimento linear da porca que se encontra acoplada a ele, resultando em movimento da mesa. Os sensores do motor enviam a informação de velocidade e posicionamento obtido ao controlador, podendo reenviar um sinal para a posição ser ajustada [27].

Pela **Figura 3.5** é possível analisar-se um tipo de mecanismo de ação direta dos eixos de translação.

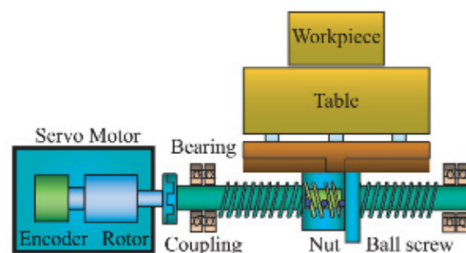


Figura 3.5: Exemplo de um mecanismo de acionamento direto dos eixos de uma máquina-ferramenta [23]

Acionamento da árvore

A velocidade e o binário de corte desejados são conseguidos pela rotação da árvore, sendo importante o motor utilizado no sistema ter capacidade para obter valores elevados destas duas variáveis. A transmissão de rotação do motor ao sistema é realizada por uma correia, onde a relação entre estes dois componentes influencia a

velocidade com que a ferramenta de corte gira. A **Figura 3.6** mostra o mecanismo da árvore [23].

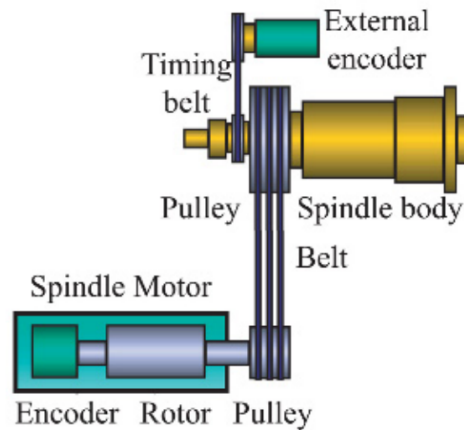


Figura 3.6: Exemplo de um mecanismo de acionamento da árvore de uma máquina-ferramenta [23]

Servomotores

Os servomotores definem-se por permitir um binário elevado, aquando de velocidades baixas e cortes pesados, e por operarem a velocidades elevadas quando se requerem movimentos rápidos. Ao mesmo tempo, respondem rapidamente, com capacidade de aceleração e desaceleração, controlando a velocidade e posição enquanto geram o movimento.

Nestas máquinas podem ser encontrados servomotores de corrente contínua (DC), corrente alternada (AC) síncrono e AC de indução, exemplificados pela **Figura 3.7** [23].

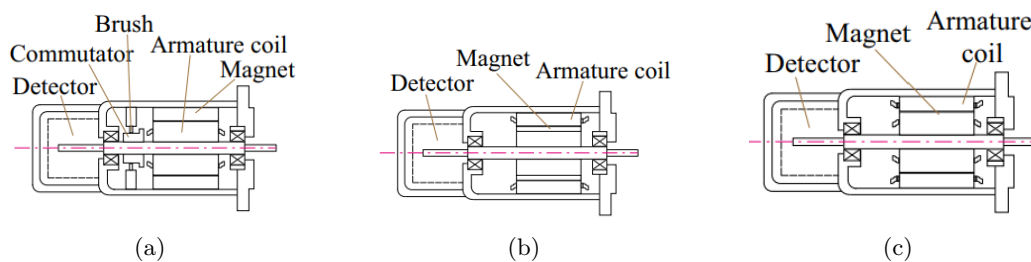


Figura 3.7: Tipos de servomotores: DC (a), AC síncrono (b) e AC de indução (c) [23]

- **Servomotor DC:** operam numa larga gama de velocidades, contudo, o seu binário máximo é limitado por haver dissipação de calor do interior das escovas, sendo esta, a sua principal desvantagem. Isto leva a que ocorram perdas mecânicas elevadas devido ao atrito entre as escovas e o comutador, levando

à manutenção mais frequente destes equipamentos. Estes motores, exemplificados na **Figura 3.7a**, como são de DC, precisam de um retificador entre o transformador e o motor para transformar a AC em DC, tornando o equipamento mais dispendioso [29] [23].

- **Servomotor AC do tipo síncrono:** são motores *brushless* (sem escovas), ao utilizarem ímãs permanentes no rotor, eliminando a necessidade de alimentação por uma corrente externa para o rotor. O estator tem uma estrutura cilíndrica com um núcleo enrolado por bobinas, alimentadas por AC em forma de ondas sinusoidais. A ausência de escovas, nestes motores, significa não haver atrito nem perdas de potência associadas ao desgaste das mesmas, o que reduz a necessidade de manutenção. Estes motores, exemplificados pela **Figura 3.7b**, têm para a mesma potência, menor volume e maior precisão em comparação com os motores DC. No entanto, devido à sua natureza, podem emitir vibrações que podem afetar o seu desempenho. O princípio de funcionamento destes motores baseia-se no conhecimento da posição do rotor, obtido através de um transdutor de posição, acoplado a ele. Esse transdutor permite definir a sequência de alimentação dos rolamentos do estator, garantindo um desalinhamento controlado entre os campos magnéticos do estator (indutor) e do rotor (induzido) para maximizar o binário. Como motores síncronos, conseguem produzir velocidade e binário constantes, mesmo em sistemas com cargas variáveis, tornando-os ideais para operações contínuas e de alta precisão [29] [23].
- **Servomotor AC do tipo indutivo:** a estrutura deste motor é idêntica à de um motor convencional, como ilustrado na **Figura 3.7c**. O estator tem enrolamentos alimentados por uma corrente trifásica, gerando um campo magnético rotativo. O rotor é composto por um eixo condutor, que pode ser do tipo bobinado. Quando a corrente trifásica é aplicada aos rolamentos do estator, cria um campo magnético rotativo que, ao interagir com o rotor, induz corrente neste último. Essa corrente induzida gera um campo magnético no rotor, que interage com o campo do estator, resultando na produção de um binário de acionamento que faz o rotor girar. Contudo, para o motor funcionar corretamente, é necessário haver uma diferença de velocidade entre o campo magnético do estator e o rotor, conhecido como "deslizamento" ou "escorregamento". Esta diferença é crucial para a gerar binário, pois, sem o deslizamento, não há corrente induzida no rotor, e, conseqüentemente, o motor não produz torque. São bastante utilizados em sistemas de controlo de movimento pela sua robustez, alta eficiência e capacidade de operar quando existem condições variáveis de carga e velocidade [23].

Para melhor compreensão das diferenças entre os motores mencionados, abaixo, na **Tabela 3.2**, é possível perceber as vantagens e desvantagens de cada motor, bem como algumas das suas características [23].

	Motor DC	Motor AC síncrono	Motor AC indução
Vantagens	Preço reduzido Ampla gama velocidade Controlo fácil	Sem escovas Fácil parar	Estrutura simples Não é necessário detetor
Desvantagens	Calor Utiliza escovas Barulho Necessário detetor posição	Estrutura complexa Ondulação de torque Vibração Necessário detetor posição	Impossível travagem dinâmica Perda de calor
Capacidade	Pequena	Pequena a media	Média a grande
Sensor	Não necessário	Codificador ou resolver	Não necessário
Vida útil	Depende das escovas	Depende dos rolamentos	Depende dos rolamentos
Velocidade elevada	Inadequado	Aplicável	Otimizável
Resistência	Fraca	Boa	Boa
Ímã permanente	Existe	Existe	Não existe

Tabela 3.2: Resumo das vantagens, desvantagens e características de cada motor [23]

Motores de passo

Os motores de passo, exemplificados na **Figura 3.8**, são utilizados quando se pretendem movimentos rápidos, repetitivos e em distâncias curtas. O funcionamento destes motores baseia-se em impulsos elétricos aplicados aos seus terminais, os quais provocam pequenos incrementos angulares, conhecidos como "passos". A direção da rotação do motor é determinada diretamente pela sequência dos impulsos elétricos, enquanto a velocidade de rotação depende da frequência desses impulsos. O ângulo total de rotação é proporcional à quantidade de impulsos aplicados.



Figura 3.8: Exemplo de motores de passo [30]

Este tipo de motores são conhecidos pela sua robustez, compactidade, baixo custo e facilidade de manutenção. Uma das grandes vantagens dos motores de passo é a sua alta capacidade de torque a baixas velocidades, eliminando, com frequência, a necessidade de redutores. Além disso, o torque de retenção - o torque máximo que o motor pode suportar sem se mover, estando com corrente - é bastante elevado, variando, aproximadamente, desde os 0,23 até aos 43,3 N.m, com capacidades de operar a velocidades de até 3000 rpm.

Contudo, apresentam, algumas limitações, significativas, quando operam a altas velocidades de rotação. Como a inércia do rotor pode ser elevada, isto pode causar dificuldades para que as bobinas do estator atraiam corretamente o rotor para a posição desejada, resultando em perda de passos e, conseqüentemente, numa perda de precisão no posicionamento. Além disso, a elevadas velocidades, estão sujeitos a efeitos de ressonância, que podem causar vibrações indesejadas e reduzir a eficiência do motor. O tempo de estabilização após a paragem ou mudança de velocidade também tende a ser elevado.

Como estes motores, normalmente, operam em malha aberta, não têm dispositivo de *feedback* para verificar se o motor atingiu a posição desejada. Quando ocorre um erro de posicionamento, como a perda de passos devido a uma carga excessiva ou a uma aceleração, ou desaceleração, rápida, o controlador não consegue detetar o erro, o que pode comprometer a precisão do sistema.

3.2.3 Acionamento linear

Algumas máquinas-ferramenta utilizam sistemas de acionamento linear para deslocar os eixos ou executar funções auxiliares, eliminando a necessidade de sistemas conversores de movimento tradicionais.

Atuadores piezoelétricos

Os atuadores piezoelétricos permitem cursos reduzidos, mas oferecem uma precisão extremamente elevada. Por essa razão, são frequentemente utilizados em máquinas que exigem movimentos finos e precisos [28].

Motores elétricos lineares

Embora apresentem um custo elevado de aquisição e manutenção, os motores elétricos lineares são utilizados para alcançar alta precisão e controlo suave do movimento. São aplicados em máquinas que necessitam de velocidades e acelerações elevadas [28].

Atuadores hidráulicos

Os atuadores hidráulicos são indicados para aplicações que requerem forças elevadas. No entanto, controlar a posição e a velocidade com estes atuadores pode ser desafiador, o que pode resultar em custos elevados, especialmente em máquinas de pequena dimensão. São, normalmente, aplicados em máquinas pesadas e robustas [28].

Atuadores pneumáticos

Os atuadores pneumáticos são os mais comuns em máquinas-ferramenta, especialmente para o acionamento de funções secundárias, como sistemas de fixação da ferramenta de corte ou da peça de trabalho, ou para a lubrificação da superfície de corte. Estes atuadores são ideais para operações que requerem movimentos repetitivos e rápidos, mas com força moderada. São de fácil manutenção e têm baixo custo de produção e baixo consumo energético [28].

3.2.4 Sistema de controlo e sensores

O controlo através de malha fechada, em equipamentos CNC, apenas se consegue devido aos sistemas de monitorização presentes nos seus componentes ou estrutura. O seu objetivo é controlar com precisão o processo de maquinar e garantir a integridade da máquina, contribuindo para se obter superfícies com elevada qualidade. Em seguida, são apresentados alguns destes componentes presentes nas máquinas CNC.

Codificadores

Os codificadores são dispositivos essenciais nos sistemas de controlo de movimento, sendo utilizados para detetar e medir a posição, velocidade e direção de um eixo de movimento, com precisão. Estes sensores de movimento geram sinais digitais conforme o movimento angular ou linear do eixo, permitindo o controlo preciso dos motores e de outros componentes do sistema.

Os codificadores podem classificar-se em lineares ou rotativos, dependendo da natureza do movimento que medem:

- *Codificadores lineares*: medem a posição ao longo de um eixo linear, sendo, habitualmente, utilizados em sistemas onde é necessário medir-se o deslocamento ou posição no eixo.
- *Codificadores rotativos*: medem a posição angular de um eixo rotativo, sendo utilizados, em motores elétricos, para o envio de informações da posição do eixo rotativo [23].

Além disso, os codificadores também podem ser do tipo ótico ou magnético, como é possível observar-se na **Figura 3.9**.

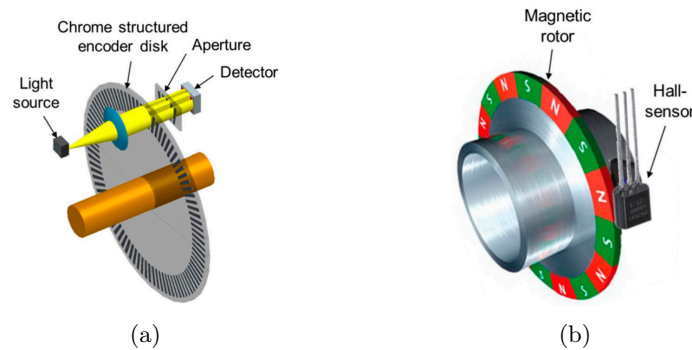


Figura 3.9: Esquema dos codificadores do tipo ótico (a) e magnético (b) [31]

A diferença entre os tipos de codificadores, mostrados na **Figura 3.9**, é o modo como é detetada a posição, contudo, ambos geram um sinal de saída de forma idêntica:

- *Codificadores óticos*: utilizam um feixe de luz que atravessa um disco com padrões, gerando pulsos que correspondem ao movimento do eixo, convertidos em sinais digitais que representam a posição e velocidade do mesmo. Destacam-se pela sua elevada precisão e resolução, sendo mais indicados para aplicar em máquinas que exigem medidas precisas.
- *Codificadores magnéticos*: detetam as mudanças no campo magnético à volta do eixo, utilizando sensores magnéticos como os *sensor hall*. Apesar de não apresentarem a mesma resolução que os codificadores óticos, são mais robustos e resistentes a condições ambientais adversas, como pó e vibrações, tornando-os adequados para ambientes industriais rigorosos [31].

Como os codificadores óticos são os mais utilizados no controlo das máquinas CNC, são abordados, em maior detalhe, seguidamente, dividindo-se em incrementais e absolutos.

- *Codificadores incrementais*: como se observa na **Figura 3.10**, são constituídos por três tipos de fendas, A, B e Z, onde as fendas A e B geram uma forma de onda de saída, enquanto a fenda Z gera a fase zero. A luz emitida pelo *light-emitting diode* (LED) é detetada por um fotodetector após passar por uma fenda do disco de rotação e por uma das fendas A, B ou Z. As fendas A e B dispõem-se com uma diferença de fase de 90° e o sinal elétrico de saída é gerado como uma onda quadrada com diferença de fase de 90° . A fenda Z gera uma onda quadrada, indicando uma revolução do codificador. Quer

isto dizer que estes codificadores geram pulsos, relativamente ao movimento do eixo, sendo que cada pulso representa um incremento do movimento. A posição exata é determinada contando-se o número de pulsos a partir duma posição de referência.

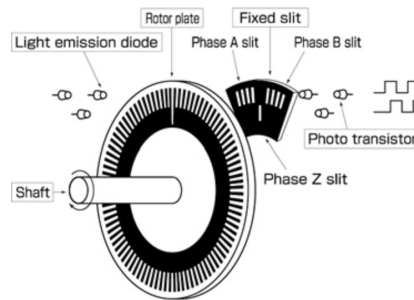


Figura 3.10: Esquema do funcionamento de um codificador incremental [32]

Este tipo de codificador é menos dispendioso e mais simples, sendo, igualmente, fácil, a transmissão do sinal pelo número reduzido de fios necessários para o efeito. O número de pulsos de saída do codificador não indica a posição absoluta de rotação de um eixo, mas sim o ângulo de rotação do mesmo. Se quisermos saber o ângulo de rotação absoluto, o número de pulsos de saída deve ser somado, e o ângulo de rotação é calculado com base no número de pulsos acumulados. Só funcionam quando é aplicada energia, caso contrário não conseguem indicar a posição. Este tipo de codificador não pode indicar a posição. Como apenas são gerados pulsos, os mesmos, à saída, têm de ser convertidos em sinais analógicos, proporcionais à frequência dos pulsos, através de um conversor como esse tipo de codificador, para a velocidade de rotação ser detetada.

- *Codificadores absolutos*: as fendas dispõem-se em círculos concêntricos, em direção ao interior do disco, os recetores são em maior número e estão dispostos de modo diferente, quando comparado com o codificador incremental. A **Figura 3.11** mostra um exemplo do funcionamento destes codificadores, onde a fenda do disco gera um bit binário e a parte mais externa do disco define-se como o bit mais baixo, existindo fendas e fotodetetores conforme o número de bits gerado. Assim, os dados de posição da rotação são fornecidos em formato binário ou decimal. Como o sinal de saída apresenta-se em bits, por serem traduzidos em código cinzento quando passam pelo LED, cada posição terá uma configuração única de bits. Isto distingue-os dos incrementais uma vez que se ocorrer interrupção ou perda de potência, não se perde a posição e não é necessário voltar a um ponto de referência. Contudo, precisam de tantos fios

de saída como o número de bits gerados, logo, apresentam maiores dimensões e têm um custo mais elevado [28].

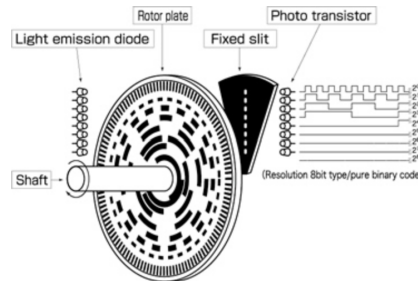


Figura 3.11: Esquema do funcionamento de um codificador absoluto [32]

Resolver

O *resolver* detecta o ângulo de rotação e a posição do motor, sendo, por isso, utilizado como um sensor. Diferentemente de um codificador, o *resolver* gera um sinal de saída analógico. A sua estrutura é semelhante à de um motor e não é sensível a vibrações ou choques mecânicos. No entanto, um dos desafios é que o sinal gerado é analógico, o que, apesar de permitir a transmissão de sinais a longas distâncias e reduzir o tamanho dos dispositivos, exige um processamento mais complexo, tornando o *resolver* mais caro em comparação com um codificador rotativo

3.2.5 Sistema de transmissão e conversão de movimento

Os movimentos realizados nas máquinas CNC devem-se a componentes específicos que constituem os sistemas de transmissão e conversão do movimento. Como o nome indica, o sistema de transmissão de movimento é responsável por este aspeto, gerado pela fonte de alimentação, enquanto o sistema de conversão, ao receber o movimento, dedica-se em convertê-lo para o tipo de movimento desejado.

Guias lineares

As guias lineares visam aumentar a estabilidade e absorver as cargas durante os movimentos da máquina, garantindo a precisão no posicionamento da peça ou do mecanismo. Elas são elementos de apoio pelos quais as diversas partes da máquina realizam o movimento, desempenhando um papel crucial ao assegurar rigor e suavidade no movimento, além de certificar a precisão e a robustez do sistema. Estes componentes também facilitam o acesso para manutenção, contribuindo para a eficiência e a durabilidade da máquina.

É possível encontrar-se alguns tipos de guias nestas máquinas, dividindo-se em três tipos [27]:

1. *Guias de deslizamento*: utilizam superfícies de deslizamento para permitir que o movimento linear aconteça, ou seja, o deslizamento acontece pelo contacto direto entre as superfícies, como se observa na **Figura 3.12**. Isto pode provocar algum atrito entre as partes móveis. São de mecanismo simples, permitem uma elevada capacidade de carga e são robustas [28];

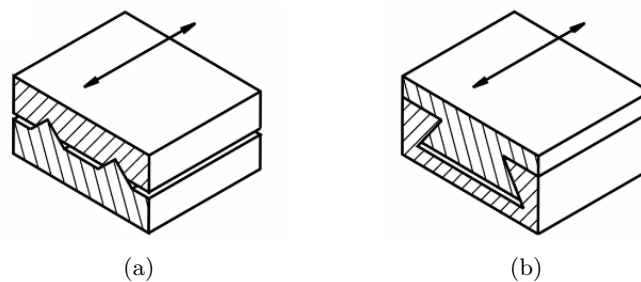


Figura 3.12: Exemplos de guias de deslizamento: prismática (a) e rabo de andorinha (b) [28]

2. *Guias de elementos rolantes*: o deslizamento acontece por meio de rolos, agulhas ou esferas, visíveis na **Figura 3.13**, para facilitar o movimento linear e reduzir o atrito, não havendo contacto direto entre as partes móveis. Aplicam-se em projetos onde é necessária sensibilidade no posicionamento, repetibilidade de movimento e velocidades elevadas [28].

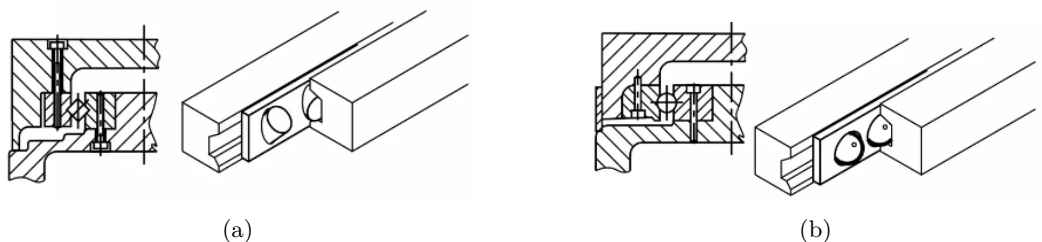


Figura 3.13: Exemplos de guias de elementos rolantes: rolo (a) e esfera (b) [28]

3. *Guias de pressurização externa*: também designadas de hidrostáticas ou guias de ar, utilizam uma camada fina de fluido pressurizado (ar, água ou óleo) entre as superfícies de contacto, diminuindo assim o atrito e permitindo um movimento suave e preciso entre as superfícies. Aparece ser o sistema ideal, dada a elevada rigidez, uniformidade do movimento, baixo atrito e boa capacidade de dissipação do calor. No entanto, o facto de necessitar de uma manutenção regular e cuidada, para além do rigor de fabrico, faz com que a sua aplicação

seja menos viável do ponto de vista económico, tornando-as mais dispendiosas [28].

Sistemas de transmissão de movimento

A transmissão de uma máquina CNC, pode acontecer diretamente ou por dimensionamento. A primeira começa quando os motores são acionados, pois os mesmos encontram-se ligados diretamente ao fuso e imprimem rotação ao sistema de transmissão, transformando o movimento angular dos motores em movimento linear. A segunda, ocorre através de uma cadeia cinemática. A primeira opção é melhor no sentido em que evita perda de potência ou folgas, contudo o seu preço é mais elevado. Por outro lado, o dimensionamento pode ser rigoroso, podendo ser difícil conseguir um posicionamento correto das ferramentas de corte [28].

As cadeias cinemáticas, essenciais para a transmissão de movimento nestas máquinas, podem ser compostas por uma variedade de elementos mecânicos, sendo os mais utilizados os seguintes:

- Trens de engrenagens: são conjuntos de engrenagens, como se pode ver na **Figura 3.14** que permitem a transmissão de movimento com alta precisão e eficiência, ideais para situações que exigem relações de transmissão fixas e robustez.



Figura 3.14: Exemplo de um trem de engrenagem [30]

- Parafuso sem-fim: visível na **Figura 3.15**, oferece grande redução da velocidade com uma relação de transmissão elevada, sendo, habitualmente, utilizado em situações que exigem controlo preciso de movimento sem haver movimento indesejado.



Figura 3.15: Exemplo de um parafuso sem-fim [30]

- **Pinhão-cremalheira:** converte o movimento rotativo em linear, sendo comum em aplicações que exigem força elevada e um curso de movimento linear extenso. Na **Figura 3.16** é possível ver um exemplo deste tipo de transmissão de movimento.



Figura 3.16: Exemplo de um pinhão-cremalheira [30]

- **Correia e polia sincronizada:** sistema simples e flexível para transmissão de movimento, como se visualiza na **Figura 3.17a**, com a vantagem adicional de absorver vibrações. Dependendo do tipo de correia utilizada, como as planas ou dentadas, essas transmissões podem oferecer uma excelente relação entre simplicidade e funcionalidade. No entanto, para operações de usinagem a altas velocidades, as correias podem não ser a melhor escolha, devido a limitações na precisão e na capacidade de transmissão de forças elevadas.



Figura 3.17: Exemplo de uma correia (a) e polia sincronizada (b) [30]

- **Correntes:** similar às correias em princípio, mas mais adequadas para aplicações onde é necessária a transmissão de potência elevada e onde o escorregamento deve ser minimizado. As correntes, como se podem ver na **Figura 3.18**, são frequentemente utilizadas em contextos onde a precisão é crítica e a força a ser transmitida é substancial.



Figura 3.18: Exemplo de uma corrente de transmissão [30]

Sistemas de conversão de movimento

Estes sistemas convertem o movimento rotacional dos dispositivos de acionamento em deslocamento linear dos conjuntos das máquinas. Para este efeito, recorrem-se a fusos com variadas configurações, que permitem, assim, o movimento, existindo uma variedade deles [28].

- Fusos trapezoidais: têm um perfil de rosca trapezoidal, como se pode ver na **Figura 3.19**, que confere maior resistência e capacidade de carga. Devido ao seu *design*, têm maior atrito que os outros tipos de fusos, o que pode limitar a sua eficiência, mas aumentar a sua capacidade de carga. Habitualmente são simples e económicos, sendo adequados para aplicações onde a precisão não é o fator mais crítico.



Figura 3.19: Exemplo de um fuso trapezoidal [30]

- Fusos de esferas recirculantes: projetados para alta precisão e eficiência, utilizando esferas que recirculam dentro da porca, como se observa na **Figura 3.20**, reduzem, significativamente, o atrito e proporciona um movimento suave e preciso. Como são bastante eficientes, permitem velocidades maiores e menor desgaste, tornando-os ideais para aplicações que exigem alta precisão e longa vida útil.
- Fusos de rolos: combinam características dos fusos trapezoidais e dos fusos de esferas recirculantes, mas em vez de esferas, utilizam rolos que deslizam ao longo das roscas do fuso. Este sistema, visível na **Figura 3.21**, oferece maior capacidade de carga comparado com os fusos de esferas, mantendo a precisão. Através do seu *design* é possível haver uma distribuição uniforme da carga, reduzindo o desgaste e aumentando a vida útil do sistema.

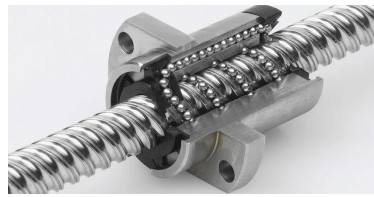


Figura 3.20: Exemplo de um fuso de esferas recirculantes [30]



Figura 3.21: Exemplo de um fuso de rolos [30]

- Fusos hidroestáticos: utilizam um filme de fluido pressurizado entre o fuso e a porca, eliminando o contacto metálico direto, resultando num movimento, praticamente, isento de atrito, minimizando o desgaste. Como não ocorre o contacto direto entre as superfícies metálicas, estes fusos conseguem suportar cargas elevadas com extrema precisão, mesmo em condições de operação intensiva.

3.2.6 Sistema de acoplamento

Este sistema é responsável pela ligação entre o eixo do motor e o fuso de esferas, sendo muito importante para a transmissão de torque e movimento entre ambos, ao garantir que ambas as partes se encontram com os eixos alinhados. Além disso, também permitem a compensação de desalinhamentos, os rolamentos dos motores ou suportes dos veios não absorvem as cargas radiais existentes causadas por estes fatores, e absorvem vibrações do sistema.

Normalmente, para fazer este tipo de ligação utilizam-se correias e engrenagens, sendo que as primeiras utilizam-se, para transmitir potência, quando o motor e o corpo do fuso estão separados, e, as segundas, são usadas quando se requer um elevado torque a uma velocidade baixa, ao atuarem como redutores de velocidade. Este tipo de utilização, chama-se "método de acionamento indireto", mas não é adequado a maquinagem de elevada velocidade, além do ruído e desgaste a longo prazo. Assim, utiliza-se o acionamento direto, onde o eixo de um motor é ligado diretamente ao corpo do fuso ou o motor é incorporado no corpo do fuso. Este método traz vantagens quando comparado com o método indireto, pois não são observadas folgas mecânicas e o desvio é menor, ao mesmo tempo que se consegue suprimir a variação do torque e rotação. A grande desvantagem é o seu preço elevado.

É possível distinguir-se dois tipos de acoplamentos, rígidos e flexíveis, exemplificados pela **Figura 3.22**. Os rígidos ligam de forma rígida dois eixos, alinhando-os para atuarem como um eixo único, não permitindo nenhum tipo de movimento relativo entre os eixos ligados, enquanto os flexíveis, interligam dois eixos enquanto acomodam certos desalinhamentos angulares, radiais ou axiais entre eles, permitindo algum grau de movimento relativo entre os eixos [23].



Figura 3.22: Exemplos de acoplamentos: rígido (a) e flexíveis (b) e (c) [30]

***Drivers* dos motores**

Os *drivers*, como se observa na **Figura 3.23** são dispositivos eletrônicos que têm o objetivo de controlar a potência e o movimento dos motores dos eixos da máquina CNC, de forma precisa e sincronizada. São responsáveis pela interpretação dos comandos provenientes do controlador CNC, ajustando, assim, a corrente e tensão enviada para os motores, com o intuito dos movimentos serem realizados com precisão.



Figura 3.23: Exemplo de um motor de passo e respetivo *driver* [30]

São várias as funções dos *drivers*, mas as mais importantes são o controlo da corrente e da tensão, a conversão dos sinais digitais e analógicos, o controlo da velocidade e posição dos motores e, por fim, proteção do motor:

- Ajustam a corrente e a tensão fornecidas ao motor para o mesmo trabalhar conforme as especificações do sistema sem o sobrecarregar ou aquecer em demasia;
- Controlam a velocidade e a posição do motor, ajustando o movimento para ir de encontro às instruções do controlador;
- Convertem os sinais digitais ou analógicos, enviados pelo controlador CNC, convertendo-os em potência que vai imprimir movimento ao motor;
- Protegem os motores através dos seus circuitos de proteção, incorporados, evitando danos no motor derivados da sobrecarga, sobreaquecimento ou falhas na fonte de alimentação.

Conforme o tipo de motor a que estão associados, existem *drivers* para motores de passo e para servomotores. Nos primeiros, ocorre o controlo o movimento dos passos, em passos discretos, fornecendo pulsos de corrente, ao motor, para o eixo ser movimentado em incrementos precisos, enquanto nos segundos, o controlo é contínuo e de alta precisão para a posição, velocidade e torque, e normalmente é utilizado um sensor de posição para ajustar o movimento, continuamente [33].

Fonte de alimentação

A fonte de alimentação, exemplificada pela **Figura 3.24** é a parte responsável por enviar a energia elétrica para todas as partes eletrónicas constituintes de uma máquina CNC, como os motores, *drivers* e o controlador CNC.



Figura 3.24: Exemplo de uma fonte de alimentação [30]

A fonte de alimentação numa máquina CNC é responsável por fornecer energia elétrica de forma estável e segura a todos os componentes eletrónicos, como motores,

drivers ou controlador CNC, convertendo a energia da rede elétrica em níveis apropriados de tensão e corrente para cada componente, garantindo que todos funcionam corretamente.

Existem diferentes tipos de fontes de alimentação, como as lineares e as chaveadas, cada uma com as suas vantagens e desvantagens em termos de eficiência, tamanho e complexidade. Além de fornecer energia, a fonte também protege o sistema contra sobrecargas, picos de tensão e outros problemas elétricos, garantindo a durabilidade e o desempenho confiável da máquina [23].

3.3 Componentes do sistema CNC

Os sistemas CNC são compostos por três unidades principais: a unidade NC, a unidade do motor e, ainda, a unidade do *driver*. Em termos práticos, apenas a unidade NC é o sistema CNC, sendo que é aqui que a interface do utilizador e o controlo da posição do sistema se encontram.

Em termos funcionais, os sistemas CNC são constituídos por três unidades: a unidade *Man-Machine Interface* (MMI), a unidade *Numerical Control Kernel* (NCK) e a unidade *Programmable Logic Control* (PLC), que se encontram interligados, como se observa na **Figura 3.25**. Em seguida, nos **Capítulos 3.3.1 a 3.3.3**, estas unidades são diferenciadas [23].

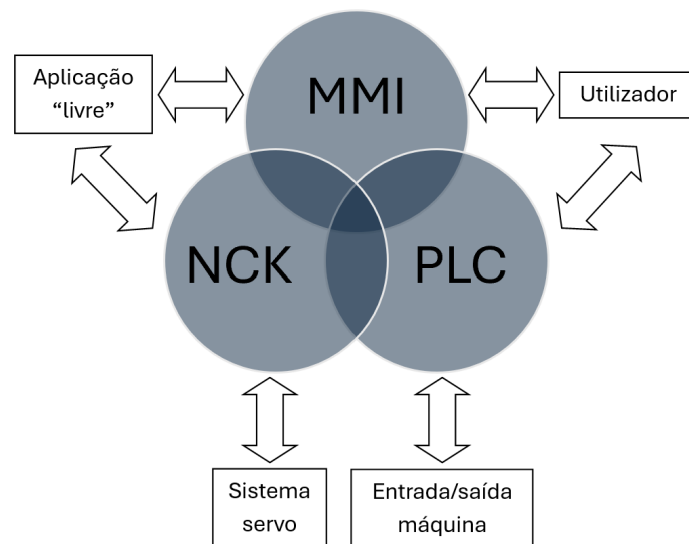


Figura 3.25: Relação entre os sistemas CNC [28]

Por outro lado, pela perspectiva de *hardware*, estas máquinas consistem no CNC, num sistema de acionamento do motor e nas próprias máquinas-ferramenta. A posição do eixo é enviada para o sistema de acionamento do motor, imprimindo

movimento aos motores, controlando a velocidade e o torque, pela transmissão de potência.

Os módulos de processamento que realizam as funções da MMI, do NCK e do PLC, são constituídos por um processador principal, uma *read-only memory* (ROM) do sistema e uma *random-access memory* (RAM) que armazena as aplicações do utilizador, os programas das peças e os programas PLC, respetivamente. Este módulo de processamento encontra-se ligado a uma interface, que permite a interação entre a máquina e o utilizador, à semelhança de um PC.

Além do mencionado, estes sistemas também são constituídos por um dispositivo *input-output* (I/O) *analógico/digital* (D/A) que realiza a comunicação direta com máquinas externas, e uma interface de comunicação para liga um dispositivo de acionamento de motor externo a um módulo I/O.

Quando se fala em *software*, os sistemas CNC, consistem em funções MMI que suportam a operação do utilizador, a edição de programas e exibem o estado da máquina. Também são funções NCK que realizam a interpretação, interpolação e controlo e, por fim, funções PLC que executam os programas de lógica sequencial [23].

A **Figura 3.26** mostra um esquema ilustrativo da relação entre a parte do *hardware* e *software* encontrada num sistema CNC.

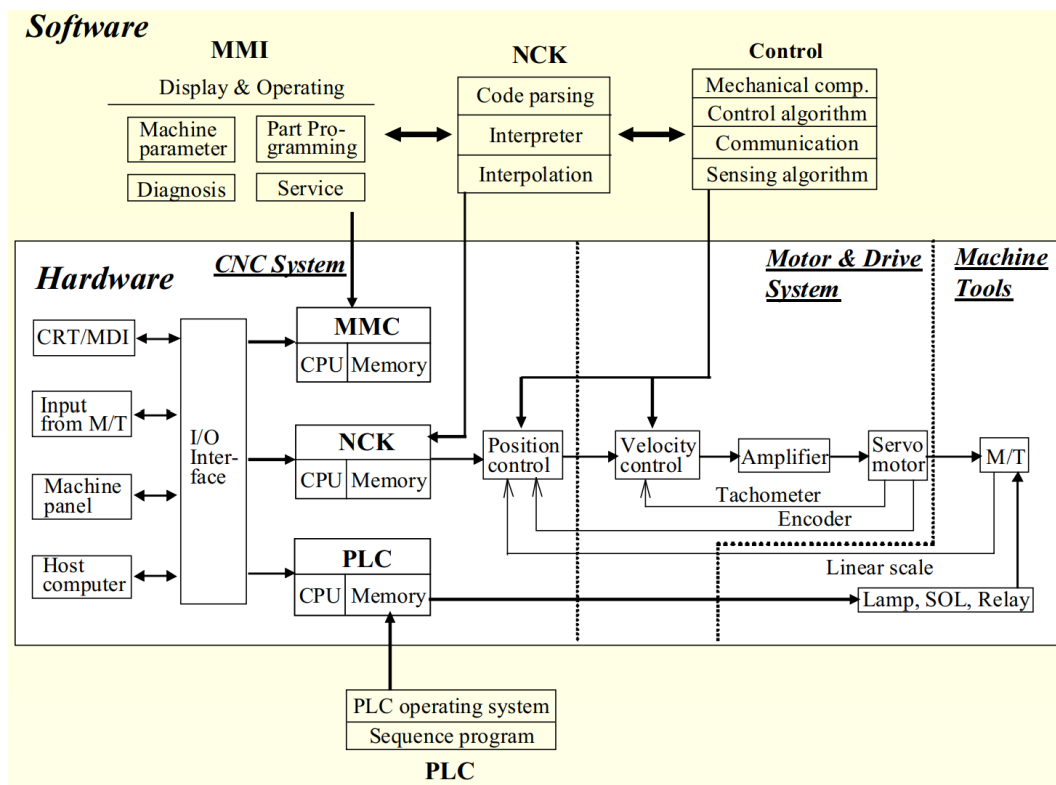


Figura 3.26: Componentes de um sistema CNC [23]

3.3.1 *Man-Machine Interface* (MMI)

A unidade MMI, exemplificada na **Figura 3.27**, é onde se realiza a comunicação entre o utilizador e a unidade NC, funcionando como uma interface, sendo aqui possível executar os diversos comandos de operação que a máquina realiza e, igualmente, a edição dos vários programas NC para maquinagem da peça.



Figura 3.27: Exemplo de uma unidade MMI, que mostra algumas funções programáveis [34]

É na MMI que se faz a visualização do estado da máquina e onde se realizam todo o tipo de comunicações entre os sistemas, permitindo a realização de testes de diagnóstico e funções de manutenção. De entre as funções desta unidade destacam-se as seguintes:

1. Funções de operação: utilizadas com frequência, ao oferecer suporte às operações de maquinagem, bem como apresentar o estado da máquina, em tempo real.
2. Funções de parametrização: são funções que parametrizam a utilização interna do sistema CNC, como, por exemplo, parâmetros da máquina, parâmetros de programação de maquinagem e parâmetros de personalização, entre o utilizador e a máquina.
3. Funções de edição de programas: editam e modificam a programação realizada para a maquinagem da peça, com base em G-code.
4. Funções de monitorização e alarme: informar o utilizador do estado da máquina e, caso necessário, realizam ações para corrigir, eventuais, problemas na mesma.
5. Funções de serviços: funções extra, presentes no sistema, que têm o objetivo de auxiliar o utilizador, como, por exemplo, serviço de cópia de parâmetros internos para o exterior e a função de comunicação, onde computadores da rede podem comunicar entre si [23].

3.3.2 *Numeric Control Kernel (NCK)*

O NCK denomina-se o núcleo do sistema CNC, sendo o controlador do sistema e, por isso, é responsável por controlar os eixos principais de movimento da máquina. Esta unidade realiza a interpretação do programa NC, criando assim a trajetória para a maquinagem das peças, através da interpolação dos vários pontos da peça, efetuando o controlo da posição dos eixos e a compensação de erros, com base no programa inserido.

É aqui que é controlado o sistema servo para que a maquinagem da peça aconteça, tendo esta unidade as seguintes funções:

1. Interpretar o programa, através do processamento e armazenamento dos dados do mesmo, para a maquinar a peça e calcular a posição para onde os eixos se devem movimentar, evitando que a máquina pare entre as várias operações. Isto é realizado recorrendo a um *buffer* interno para otimizar o fluxo de dados;
2. Interpolar as linhas, círculos ou parábolas, calculando a posição e velocidade dos eixos da máquina CNC, produzindo pulsos que determinam a trajetória da ferramenta com base nos dados armazenados. A precisão da máquina é definida pela quantidade de deslocamento por pulso;
3. Controlar a aceleração e desaceleração, através dos dados interpolados, para harmonizar o movimento da máquina, minimizando a ocorrência de vibrações mecânicas e choques durante o processo de maquinagem das peças. Este controlo tanto pode ser aplicado antes como depois da interpolação;
4. Controlar a posição dos eixos, na maioria das vezes através de um controlador *Proportional-Integral-Derivative* (PID), que ajusta a velocidade do motor, para alinhar a posição real da ferramenta de corte à posição desejada, reduzindo estes erros de posicionamento pela compensação de eventuais erros derivados da interpretação do programa [23].

3.3.3 *Programmable Logic Control (PLC)*

A unidade PLC, controla, sequencialmente, a troca de ferramentas, a velocidade do *spindle*, a troca das peças de trabalho e o processamento de sinais de entrada e saída, controlando a máquina, com a exceção do sistema servo, executando programas lógicos, em sequência.

Esta unidade encontra-se em constante troca de informações com o NCK para garantir a operação eficaz da máquina. As suas vantagens são as seguintes:

- Flexibilidade;
- Escalabilidade

- Eficiência económica;
- Redução de tamanho;
- Fiabilidade;
- Elevado desempenho.

Na **Figura 3.28** é possível observar-se a lógica de funcionamento da unidade PLC [23]. Basicamente, nesta unidade, é criado e compilado um programa, para uma determinada tarefa, num dispositivo específico de nome programador. Este programa, após ser convertido para a linguagem de interpretação de um PLC, é enviado para o módulo *Central Processing Unit* (CPU). Ao mesmo tempo que isto acontece, o estado do PLC é enviado para o programa PLC, para a sua atividade ser monitorizada.

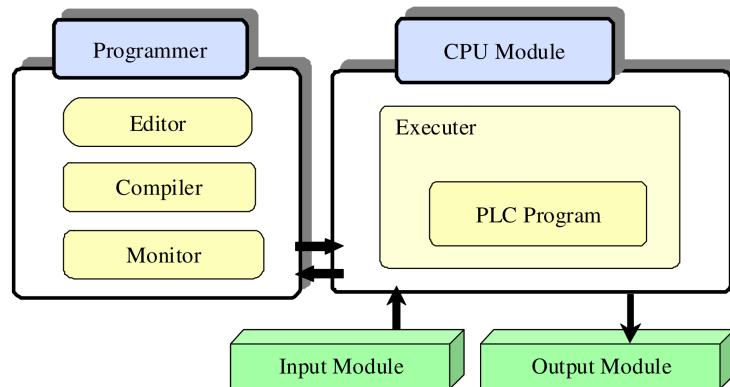


Figura 3.28: Arquitetura e função do sistema PLC [23]

O módulo que lê o programa e o executa, sequencialmente, em operações lógicas, denomina-se executor, sendo o núcleo do *kernel* PLC. O executor lê, constantemente, os pontos de entrada, realizando as operações lógicas do programa, e envia os comandos para os pontos de saída.

3.4 Utilização de máquinas CNC em ortoprotesia

Com a compreensão das máquinas CNC e dos seus componentes, é possível explorar como estas tecnologias são aplicadas na maquinagem de diferentes materiais. Como referido no **Capítulo 2**, é cada vez mais frequente a introdução de novas tecnologias na área da ortoprotesia, com intuito de diminuir tempos de produção e obter melhores resultados nos dispositivos médicos - próteses e ortóteses.

As tecnologias digitais, têm dado resposta a várias questões resultantes da parte prática do fabrico de próteses e ortóteses. Com a introdução da digitalização, das partes do corpo, através de *scanners* 3D, bem como a introdução de tecnologia

CAD/CAM e do fabrico aditivo, houve uma melhoria, significativa, na funcionalidade destes dispositivos.

A par com as tecnologias referidas, foi sendo introduzido o fabrico de moldes positivos com recurso à tecnologia CNC, quer pela utilização de máquinas-ferramenta CNC, quer por braços robóticos ou outro tipo de dispositivo que permita o fabrico dos moldes positivos para esta área. Isto acontece após o *design* do dispositivo ou molde estar concluído e serem definidas as trajetórias, habitualmente, recorrendo a G-code, para o fabrico dos moldes.

Existe uma propensão em ser utilizado o fabrico aditivo para este tipo de dispositivos, no entanto, este tipo de tecnologia traz alguns desafios, para esta área, nomeadamente quando se pretende obter dispositivos de grande escala. Os problemas prendem-se, essencialmente, com a velocidade de produção do dispositivo e com as propriedades mecânicas do produto final, devido ao volume de produção ocupado ser significativo, da espessura das paredes ser pequena, e por terem grandes dimensões e, ao mesmo tempo, serem de estrutura leve. Além disso, o preço da tecnologia aditiva, neste caso, impressoras 3D, é alto e o tamanho físico destes sistemas ser bastante elevado, quando é necessário obter-se dispositivos de grande escala.

Face aos desafios enumerados, existe, igualmente, uma tendência em serem utilizados braços robóticos nesta área, especialmente quando se pretendem obter dispositivos grandes, como as ortóteses ou encaixes para as próteses, ao ser possível obter moldes de grande escala com esta tecnologia. Apesar de apresentar bastante flexibilidade a nível de produção, pelas suas características únicas, ao permitirem produzir moldes com formas complexas e de elevado detalhe, o maior problema encontrado, nos robôs, é o facto de eles exigirem a utilização de *hardware* adicional, como a mesa rotativa. Ainda, outros aspetos negativos da sua implementação, estão relacionados com o preço da estrutura, tal como no fabrico aditivo, e, ainda, com a complexidade, significativa, da melhor estratégia para serem obtidas as melhores trajetórias para se obter os moldes positivos, o que é de extrema importância à aplicação nesta área, que exige que os processos sejam rápidos e simples.

Assim, torna-se evidente que a abordagem mais simples para esta área, para a obtenção de moldes positivo, passa pelo recurso a máquinas especiais, que utilizam a tecnologia CNC, onde as suas características e funcionalidades são adaptadas aos requisitos específicos desta área. Aqui, o único problema da sua implementação é o custo inicial ser, igualmente, elevado.

Ao considerar o desempenho das máquinas CNC, na área da ortoprotesia, e as necessidades existentes em pequenas empresas, é importante referir alguns fatores, para a máquina ser adequada à prática:

- Preço favorável;
- Suficientemente rápida e funcional;

- Pequenas dimensões;
- Robusta;
- Ergonomia adequada;
- Treino simples para o utilizador;
- Fácil manuseio do material e limpeza dos resíduos;
- Adequada para mudanças de localização no espaço de trabalho;
- Construída de modo que a sua manutenção seja fácil;
- Silenciosa;
- Eficiente em termos de energia;
- Segura.

Existe uma grande variedade de máquinas para a maquinação de moldes positivos, mas, normalmente, os materiais utilizados, para o efeito, são blocos de PU expandido, espumas de acetato de etileno-vinil (EVA) ou poliestireno, por serem os materiais relativamente macios e maioritariamente utilizados nesta área, quer para a obtenção de moldes positivos, quer para o fabrico dos dispositivos médicos. Estas características também reduzem significativamente as exigências de rigidez e potência das máquinas-ferramenta CNC quando comparadas com as mesmas utilizadas no corte de metais [35].

Quando são analisadas as configurações típicas de máquinas CNC para utilização na área da ortoprotesia a questão é complexa, nomeadamente do ponto de vista dos requisitos, nomeadamente o preço de implementação destas máquinas. Os *designs* existentes são escassos, podendo ser isto explicado por, maioritariamente, existirem soluções comerciais protegidas [35].

Muitas vezes, são encontradas máquinas personalizadas, conforme o que os centros ortopédicos pretendem obter com as mesmas. Assim, tipicamente, encontram-se máquinas **fresadoras** e **torneadoras**, pois o grande objetivo é fresar e tornear os blocos de matéria-prima, para se obter a peça final.

A seguir, são mostradas as diferenças entre estes dois tipos de máquinas e, mais adiante, no **Subcapítulo 3.6**, enumeram-se as diferenças entre os tipos de maquinação que estas máquinas realizam.

3.4.1 Máquinas fresadoras

Uma máquina fresadora é uma máquina com a capacidade de realizar movimento e corte, recorrendo a uma fresa, ao longo de pelo menos dois eixos, simultaneamente.

Podem variar em tamanho, características, adequação para certos trabalhos, mas todas têm um denominador comum que é os seus eixos principais serem o X e Y, por isso também são denominadas máquinas XY.

São vários os tipos de fresadoras existentes e podem classificar-se em três categorias - pela orientação dos eixos, pelo número de eixos, e, ainda, pela presença ou ausência de um trocador de ferramentas.

A classificação, conforme a orientação dos eixos, ocorre pela relação da posição do *spindle* à mesa de trabalho. Assim existem duas possibilidades - horizontal, como mostra a **Figura 3.29**, onde o movimento do *spindle* acontece no eixo X; ou vertical, onde o movimento do *spindle* ocorre no eixo Y. Contudo, podem ser encontradas configurações alternativas que dependem do método de controlo, tamanho, número de eixos, finalidade da fresagem e fonte de alimentação [36].



Figura 3.29: Exemplo de uma máquina fresadora horizontal [37]

Grande parte das máquinas CNC existentes para a área da ortoprotesia, são construídas com um *design* de três ou quatro eixos, sendo que são raros os casos onde se encontram máquinas com cinco eixos de funcionamento. São modelos de máquinas semelhantes às máquinas CNC para corte de metal, mas que são menos exigentes do ponto de vista mecânico [35].

3.4.2 Máquinas torneadoras

Uma máquina torneadora, ou torno, é utilizado para maquinar peças cilíndricas ou cónicas, ocorrendo a remoção de material de um bloco, normalmente, cilíndrico, através de uma ferramenta de torneamento para corte externo. O torno pode, igualmente, ser utilizado para outras operações, como o mandrilamento ou para fazer ranhuras ou roscas, caso a ferramenta de corte utilizada seja adequada.

Este tipo de máquinas pode ser classificado pelo seu *design* e pelo número de eixos, sendo que a configuração habitual é o torno vertical ou o horizontal, e este último é o mais habitual [36]. A **Figura 3.30** exemplifica um torno vertical.

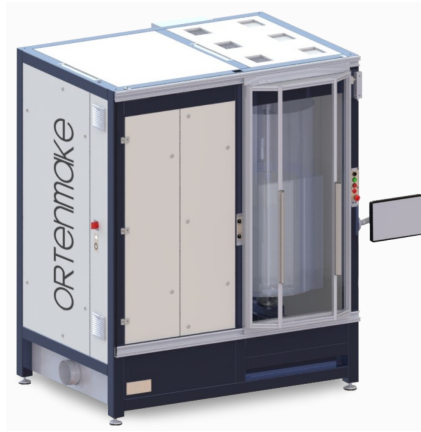


Figura 3.30: Exemplo de um torno vertical CNC [16]

Habitualmente, a grande diferença entre os tipos de máquinas de torno é o número de eixos programáveis. Os tornos CNC verticais são normalmente compostos por dois eixos, enquanto os horizontais, mais comuns, encontram-se projetados com dois, três, quatro ou seis eixos programáveis, facilitando o fabrico de peças mais complexas. Além destas configurações de eixos, é, ainda, possível serem encontradas máquinas torneadoras com a mesa inclinada, permitindo certos tipos de fabrico, igualmente, desafiadores.

Como referido, o método mais comum para definir este tipo de máquinas é pelo seu número de eixos. Assim, tipicamente, estas máquinas CNC são projetadas com dois eixos padrão, X e Z, perpendiculares entre si, que realizam os movimentos de corte nestes dois eixos. Ou seja, o eixo X realiza o movimento transversal da ferramenta de corte, enquanto o eixo Z realiza movimento longitudinal. Num torno horizontal, a orientação dos eixos ocorre com o eixo X a movimentar-se para cima e para baixo, enquanto o eixo Z movimenta-se para a direita e esquerda [36].

É possível encontrar alguns casos de máquinas CNC de torno onde ocorre a fixação de um eixo, normalmente o de rotação, e o outro eixo, longitudinal, é posicionado colinearmente ao eixo de rotação, sendo que o bloco é preso a esse eixo. A fixação do bloco ocorre num prato de fixação através de pinos ou parafusos, promovendo uma melhor fixação do bloco. O eixo fixo, tanto pode ser montado horizontal como verticalmente, e neste último, existem tipos de fixação diferentes onde o bloco pode ser fixado na parte superior, reduzindo as vibrações durante a maquinagem da peça.

O problema da utilização deste tipo de máquinas é a área de corte ser limitada, não atingindo certas zonas da peça, o que pode comprometer o resultado da peça [35].

3.5 Tecnologia existente no mercado

Com o evoluir da tecnologia CNC, surgiram cada vez mais máquinas dedicadas a essa finalidade. Assim, vários fabricantes desenvolveram máquinas CNC direcionadas para a área da ortoprotesia, variando em escala, conforme o tamanho da peça que se pretende obter.

As máquinas de maior dimensão são utilizadas para a maquinação de peças grandes, como moldes positivos para o fabrico de próteses (encaixes) e ortóteses (coletes e aparelhos de marcha longos), utilizando blocos de PU de média e alta densidade para esse propósito. Por outro lado, quando se pretende obter peças menores, como palmilhas, utilizam-se espumas de PU ou de EVA, sendo que, para essas aplicações, as máquinas CNC são de dimensões menores.

As máquinas CNC disponíveis e dedicadas à maquinação de peças na área de próteses e ortóteses, geralmente, vêm equipadas com um *software* CAM próprio, responsável por gerar os caminhos de corte, os *toolpaths*, que a máquina e a ferramenta de corte devem seguir para obter a peça.

Outro pormenor importante destas máquinas é que o controlador CNC já vem embutido nelas, facilitando ainda mais a operação e integração dos sistemas. Este controlador é responsável por executar os comandos gerados pelo *software* CAM, coordenando os movimentos da máquina com precisão.

Em conjunto com o controlador, estas máquinas também vêm equipadas com uma interface de utilizador intuitiva, permitindo a configuração e monitorização dos processos de maquinação eficientemente.

Com o intuito de tornar a máquina o mais económica possível, tanto na construção como no controlo, muitas empresas recorrem a *softwares* CAM *open source* para gerar o G-code e a controladores do mesmo tipo, deixando para uma fase posterior a decisão sobre a implementação de uma interface de utilizador para o controlo da máquina.

Contudo, em alguns casos, como no exemplo da máquina CNC em estudo, há empresas que, devido a um menor volume de trabalho ou capacidade financeira limitada, optam por construir as suas próprias máquinas CNC, utilizando, para isso, estruturas *standard* disponíveis no mercado. A principal contrapartida desta abordagem é a necessidade de implementar todos os componentes de *hardware* e *software* necessários para garantir a operação correta da máquina e a maquinação das peças. Desta forma, com o objetivo de tornar a máquina o mais económica possível, tanto na construção como no controlo, muitas empresas recorrem a *software* CAM *open source* para gerar o G-code e a controladores do mesmo tipo, que são igualmente acessíveis, quando comparadas com as opções disponíveis no mercado. Esta escolha permite uma significativa redução de custos, mas também exige decisões importantes quanto à implementação de uma interface de utilizador eficaz para o

controlo da máquina, garantindo que o sistema seja fácil de utilizar e ofereça as funcionalidades necessárias para uma operação eficiente e precisa.

Abaixo, nos **Capítulos 3.5.1 a 3.5.4**, são apresentadas máquinas CNC existentes no mercado, bem como opções de *softwares* CAM e controladores CNC *open source*, quando se pretende utilizar soluções mais económicas.

3.5.1 Máquinas-ferramenta CNC

Existem várias empresas que desenvolvem máquinas CNC para esta área, as quais variam conforme as dimensões da peça a maquinar, os eixos de atuação que as mesmas permitem e o tipo de material que permitem maquinar. Assim, podem ser encontradas no mercado, máquinas com 2, 3 e 4 eixos de atuação, que permitem a maquinação de espumas de PU ou EVA e, ainda, em alguns casos, cortiça [35].

Em seguida são apresentados alguns exemplos de máquinas existentes no mercado que se encontram dedicadas e desenhadas para a área da ortoprotesia.

Rodin4D Model SX

A máquina observada na **Figura 3.31a**, desenvolvida pela empresa Rodin4D, tem três eixos, dois lineares e um de rotação. Ela é controlada em tempo real através de uma interface gráfica de fácil utilização e tem um sistema de bloqueio automático das portas, durante o processo de maquinação, garantindo a proteção do operador contra as peças em operação. É compatível com diversos *softwares* CAM, permitindo a utilização de ficheiros em diferentes formatos, como *.stl*.

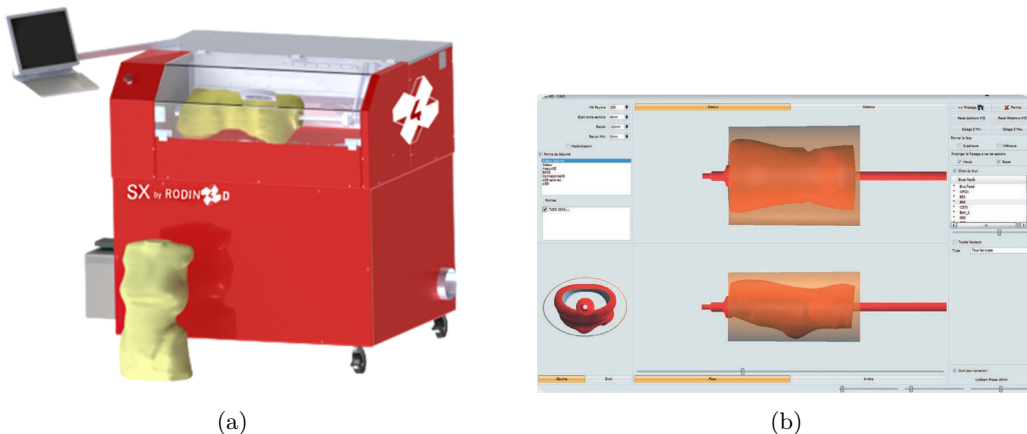


Figura 3.31: Máquina CNC Model SX (a) e *software* CAD/CAM Neo CAM, desenvolvidos pela empresa Rodin4D (b) [38]

Esta máquina encontra-se especialmente desenhada para a maquinação de blocos de PU, permitindo um volume médio da peça maquinação. Deste modo, podem ser

obtidos moldes positivos para, por exemplo, o fabrico de coletes, próteses do membro inferior e membro superior, capacetes, entre outros.

Além de desenvolver máquinas com tecnologia CNC, a Rodin4D também oferece *software* CAD e CAM. O *software* apresentado por eles, chamado Neo CAM, facilita a programação das máquinas CNC, oferecendo uma interface amigável para o utilizador, que permite visualizar a simulação da maquinagem da peça, como ilustrado na **Figura 3.31b**. Este *software* permite a parametrização da máquina, como se pode observar na **Figura 3.31b**, incluindo as dimensões do bloco de PU e as especificações de maquinagem conforme o objetivo pretendido. Outra vantagem deste *software* é a possibilidade de reutilizar um bloco de PU previamente maquinado, minimizando os custos de matéria-prima [38].

Na **Tabela 3.3** apresentam-se algumas das características que podem ser encontradas neste equipamento.

Tabela 3.3: Características e especificações da máquina CNC Model SX da empresa Rodin4D [38]

Dimensões e estrutura	Tamanho	1500*1200*1400 mm
	Peso	330 kg
	Volume máximo peça	420 mm diâmetro / 660 mm altura
	Alcance horizontal máximo (eixo R)	220 mm
	Alcance vertical máximo (eixo Z)	700 mm
	Capacidade maquinagem	300 mm diâmetro / 450 mm altura
	Ruído	80 db (com sucção ativa)
	Ferramenta fixação blocos PU	
Capacidade corte e maquinagem	Velocidade operação	Até 330 mm/s
	Velocidade eixo	1000 a 10000 rpm
	Ferramenta corte	190 mm comprimento / 15 mm diâmetro
Materiais compatíveis	Blocos PU com densidade máxima de 200 kg/m³	
Sistema controlo e transmissão	Spindle	Esférico
	Ligação à internet	Wi-Fi e RJ45
	Sistema operativo	Windows 7
Sistema elétrico	Fonte alimentação	220 V / 110 V
	Consumo energia	50 Hz / 10 A / 2,2 kW

Vulcan TWINCAM

A Vulcan TWINCAM, visível na **Figura 3.32**, é uma máquina fresadora, disponível nas opções de três ou quatro eixos de atuação, especialmente concebida para a produção em série de palmilhas, desenvolvida pela empresa Sensor Medica. A versão de três eixos está equipada com um *spindle* elétrico enquanto a versão de quatro eixos tem dois *spindles* independentes, permitindo maior flexibilidade e eficiência no processo de maquinagem.



Figura 3.32: Máquina CNC Vulcan TWINCAM, da empresa Sensor-medica [39]

Uma das principais diferenças relativamente às máquinas CNC apresentadas anteriormente é o tamanho compacto da Vulcan e as dimensões das peças finais que ela pode produzir. Sendo uma máquina, significativamente, menor, ela pode ser facilmente colocada numa bancada de laboratório, o que a torna ideal para a fabricação de peças de pequenas dimensões, como palmilhas personalizadas.

Além disso, a Vulcan vem equipada com um sistema de aspiração integrado, que remove eficientemente o pó e outros resíduos gerados durante a maquinagem, mantendo o ambiente de trabalho limpo e seguro. O que realmente diferencia a Vulcan de outras máquinas CNC, com objetivos semelhantes, é o seu sistema de vácuo para fixação dos blocos de espuma, na mesa de trabalho. Este sistema elimina a necessidade de usar fita adesiva dupla face, comum em outras máquinas, proporcionando uma fixação mais rápida, segura e sem resíduos, além de facilitar a remoção e reposicionamento dos blocos.

Outra característica notável da Vulcan é a sua interface de utilizador intuitiva, que permite uma configuração fácil e rápida dos parâmetros de maquinagem, como as dimensões do bloco de espuma e as especificações detalhadas do corte. Com a possibilidade de personalizar o processo de maquinagem, a máquina consegue otimizar o tempo de produção e garantir alta precisão na fabricação das palmilhas.

Adicionalmente, a versão de quatro eixos oferece maior versatilidade ao permitir a maquinação de formas mais complexas e detalhadas, ampliando as possibilidades de *design* e personalização das palmilhas produzidas.

O *software* com que a máquina Vulcan opera é compatível com qualquer projeto de palmilhas no formato *International Society Organization* (ISO) G-code [39].

Em seguida, na **Tabela 3.4**, são apresentadas algumas das características que podem ser encontradas na máquina Vulcan TWINCAM, de quatro eixos.

Tabela 3.4: Características e especificações da máquina CNC Vulcan TWINCAM da empresa Sensor Medica [39]

Dimensões e estrutura	Tamanho	820*1110*1070 mm
	Peso	200 kg
	Alcance vertical máximo (eixo Z)	80 mm
	Capacidade maquinação	400*300*80 mm
	Sistema de vácuo para fixação dos blocos de espuma	
Capacidade corte e maquinação	Velocidade operação	Até 250 mm/s
	Velocidade eixo	24000 rpm (motor assíncrono)
	Aceleração	Até 2800 mm/s ²
	Ferramenta corte	Aço. Precisão de 0,05 mm
Materiais compatíveis	Blocos PU e EVA	
Sistema controlo e transmissão	Spindle (2)	Elétrico de 24000 g/min
	Ligação ao PC	<i>Ethernet</i>
	Sistema operativo	Windows 7
Sistema elétrico	Fonte alimentação	220 V / 110 V
	Consumo energia	14,10 A

3.5.2 Controladores e MMI para máquinas CNC

Nos últimos anos, com o avanço das tecnologias CNC e a crescente utilização de máquinas-ferramenta que empregam essa tecnologia, houve um foco na simplificação das interfaces do utilizador, bem como na implementação de *hardware* e formatos de ficheiros para controlar os sistemas de maquinação.

Como os controladores e as interfaces do utilizador estão frequentemente interligados, com a parte do controlo acoplada à interface, a aquisição dos restantes componentes dessas máquinas CNC - motores, *drivers*, sensores, entre outros - acaba por ser uma tarefa complexa e dispendiosa. Como mencionado anteriormente, este fator torna todo o sistema dispendioso, limitando a adoção e implementação dessas tecnologias.

A interface do utilizador, normalmente, é habilitada com um ecrã sensível ao toque ou painéis de controlo que mostram informações em tempo real sobre o estado da máquina, a posição das ferramentas e o progresso do trabalho. A integração completa do controlador e da interface simplifica o processo de configuração e operação da máquina, bem como garante a realização de ajustes rápidos e precisos, por parte do utilizador, aumentando a produtividade e reduzindo o risco de erros durante a produção.

Os controladores têm algoritmos avançados de controlo de movimento, garantindo que a trajetória da ferramenta segue exatamente o planeado no *software* CAM, têm em consideração fatores como a aceleração, desaceleração e compensação de erros. Estas características são cruciais para manter a alta qualidade das peças maquinadas, especialmente em aplicações sensíveis como a fabricação de ortóteses e próteses, onde a precisão milimétrica é necessária.

O controlador também pode estar equipado com funções de monitorização em tempo real, que permitem ajustar os parâmetros de corte durante a operação para otimizar o desempenho, prolongar a vida útil das ferramentas e garantir a segurança do processo. Esta combinação de precisão, eficiência e facilidade de utilização faz com que as máquinas CNC com controladores embutidos sejam uma escolha ideal para a produção de peças complexas e personalizadas na área de ortoprotesia.

Hoje em dia, há uma tentativa de desenvolver controladores em tempo real utilizando uma arquitetura aberta, visto que a grande maioria dos fabricantes desse tipo de máquina, como a Siemens, Fanuc, Mitsubishi, entre outros, oferece apenas controladores com arquitetura fechada, dificultando a incorporação de controle avançado nos sistemas CNC. Os controladores dessas empresas são projetados especificamente para integração com as suas próprias máquinas, não garantindo a mesma eficiência quando aplicados em máquinas de outros fabricantes ou em projetos de menor escala [40] [41].

Como, muitas vezes, é difícil aceder a este tipo de componente, ao existirem limitações, a nível do fabricante, para a sua aquisição, a solução mais "fácil" é a aquisição de *kits retrofit* CNC para melhorar todo o sistema da máquina-ferramenta. Os grandes fabricantes das máquinas CNC disponibilizam este tipo de *kits*, como se pode observar na **Figura 3.33**, no entanto, são, igualmente, dispendiosos, o que, mais uma vez, torna todo o sistema dispendioso, rondando os 18000€. Os controladores apresentados acima, nesta **Figura** já vêm equipados com o controlador CNC, interruptor remoto *ON/OFF*, codificadores lineares, *drivers* e servomotores, além de uma consola com interface do utilizador, permitindo que a máquina CNC seja equipada com mais componentes [42].



Figura 3.33: Exemplo de *kits retrofit* CNC da Siemens (a) e Acu-Rite (b) [42]

Atualmente, existem, no mercado, alternativas *low cost* para este tipo de componentes, vendidas, a preços mais vantajosos, por empresas distribuidoras de produtos *low cost*, como a AliExpress, Alibaba ou VEVOR. O problema destes componentes é falta de garantia relativamente à instalação, eficiência e compatibilidade com a máquina-ferramenta CNC e os seus respetivos componentes. A **Figura 3.34** exemplifica um desses controladores disponíveis neste tipo de mercado [43].



Figura 3.34: Exemplo de *kit* CNC acessível através da AliExpress [43]

No caso das implementações *low cost* pode, também, haver problemas na padronização ou qualidade de fabricação dos controladores em comparação com fabricantes de renome, levando a incompatibilidades ao nível de *hardware*, como problemas de ligação com os motores, *drivers* e outros componentes críticos da máquina CNC, ou de *software*, pois habitualmente o *software* de proprietário é limitado ou com suporte inadequado, dificultando a integração com os restantes componentes, dificuldades de parametrização e falhas nas operações de maquinagem ou diminuição da eficiência do sistema, tornando o processo de maquinagem menos fiável.

Tal como acontece nos *kits retrofit* CNC de fabricantes de renome, estes *kits low cost* também vêm equipados com controlador CNC, interruptor remoto *ON/OFF*, codificadores lineares, *drivers*, servomotores, e uma consola com interface do utilizador. A diferença principal reside nos valores de aquisição desses componentes, que, no caso dos fabricantes *low cost*, anda em cerca de 600€ [43].

O problema encontrado na aquisição de componentes de diversos fabricantes para construir máquinas CNC está na compatibilidade entre o *hardware* e o *software* dos componentes, pois cada um deles necessita do seu próprio *software* de parametrização, desenvolvido pelos respetivos fabricantes. No caso das implementações *low cost*, também podem surgir problemas relacionados à padronização ou à qualidade de fabricação dos controladores, quando comparados aos dos fabricantes maiores. Isto pode levar a incompatibilidades a nível de *hardware*, como problemas de ligação aos motores, *drivers* e outros componentes críticos da máquina CNC. A nível de *software*, os controladores *low cost*, vêm, frequentemente, com *software* proprietário limitado ou com suporte inadequado, dificultando a integração com os restantes componentes. Este fator cria dificuldades de parametrização e resulta em falhas operacionais ou na diminuição da eficiência do sistema, tornando o processo de maquinação menos fiável [41].

Para colmatar os problemas encontrados, mencionados acima, surgem os controladores de arquitetura aberta para eliminar o problema de implementação nos sistemas CNC, tornando os sistemas de controlo mais flexíveis, permitindo que sejam acoplados a uma variedade de máquinas-ferramenta CNC, dando oportunidade de criar uma automação mais fácil de novos processos de fabrico e soluções CNC mais personalizadas.

Este tipo de arquitetura é uma base de protocolos de comunicação e de capacidades do sistema, confiáveis, estáveis e modulares, a nível do *software* e *hardware* e visa substituir ou complementar um controlador CNC existente, mantendo os motores de acionamento dos eixos e do *spindle* originais, bem como as interfaces do utilizador [41].

Normalmente, os controladores de arquitetura aberta são construídos utilizando computadores padrão, operam com sistemas operativos *Windows* ou *Linux*, em PC's ou estações de trabalho [40] [41]. Também utilizam tecnologias com microprocessadores e sensores acessíveis e plataformas de desenvolvimento baseadas nesta tecnologia, como Arduino, Ti LaunchPad, Teensy ou Beaglebone. Atualmente, algumas tecnologias CNC, como a impressão 3D ou maquinação CNC migraram para um *hardware open source* [41].

3.5.3 *Software* CAM para gerar G-code

Os *softwares* CAM desempenham um papel crucial, nos processos de fabrico, permitindo criar um programa de maquinagem CNC, em linguagem G-code, com base na geometria e no modelo 3D da peça, além da parametrização das operações que a máquina realizará. Embora a maioria das máquinas CNC utilize G-code como padrão, cada máquina é configurada, especificamente, para o tipo de maquinagem que realiza. Assim, devem ser realizadas algumas operações nos *software* CAM para garantir a precisão e eficiência do processo:

- Definir as formas desejadas, utilizando modelos geométricos 3D, estabelecendo a geometria inicial e final, do bloco a ser maquinado;
- Importar o modelo a partir do *software* CAD utilizado, garantindo a integridade dos dados durante a transição entre os ambientes CAD e CAM;
- Incluir informações de maquinagem, como a seleção das ferramentas adequadas, parâmetros de corte (velocidade de avanço e corte e profundidade de corte) e estratégias de maquinagem (*paths* a realizar), que determinam a trajetória que a ferramenta efetuará;
- Simular os *paths* definidos, permitindo a visualização da peça após cada operação, identificar possíveis colisões, prever problemas na maquinagem e calcular os tempos da mesma;
- Configurar o pós-processamento, gerando o G-code, no formato específico para a máquina escolhida, adaptando-o às características e limitações da máquina CNC [44].

Como mencionado, estes *softwares* possibilitam a simulação do processo de maquinagem, o que é essencial para visualizar possíveis colisões ou erros, antes da máquina iniciar a operação. Além disso, permitem a capacidade de ajustar, automaticamente, os parâmetros de corte com base nas propriedades do material a ser maquinado, contribuindo para a eficiência e segurança do processo.

São utilizados diversos *softwares* pelos fabricantes de máquinas CNC, entre os quais se destacam o MasterCam, Fusion360 ou DeskProto. Estes *softwares* oferecem soluções CAM, que variam em termos de complexidade e custo, permitindo que os utilizadores escolham a ferramenta que melhor se adapta às suas necessidades [44].

O MasterCam, visível na **Figura 3.35**, é o *software* CAM mais utilizado no mundo. Integra soluções CAD/CAM, o que permite criar o modelo 3D ou importar ficheiros provenientes de outros *softwares* CAD, em vários formatos, como .stl. Além destas características, permite criar e simular os *paths* de maquinagem, gerando o G-code, no mesmo ambiente, simplificando o fluxo de trabalho [45].

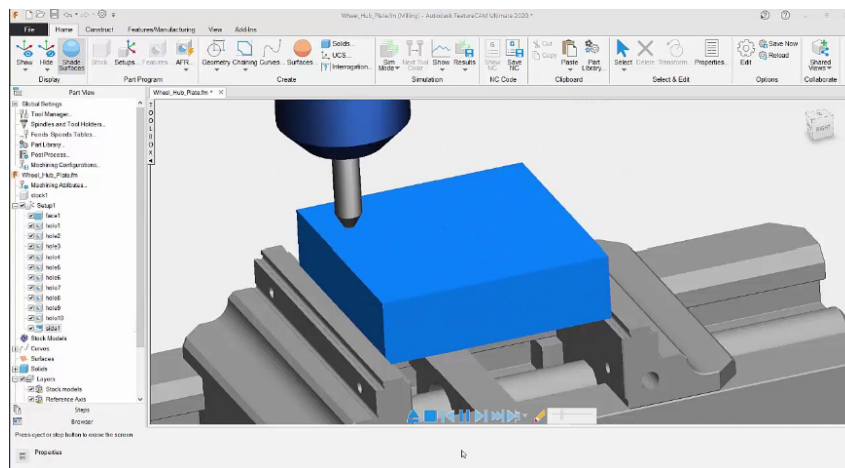


Figura 3.35: Simulação da maquinagem de uma peça no *software* MasterCam [45]

Por outro lado, o DeskProto, observado na **Figura 3.36**, é uma opção popular para quem procura um sistema CNC mais acessível, especialmente devido à disponibilidade de versões, tornando-o uma escolha atrativa para pequenos volumes de negócio. É compatível com os Sistema Operativo (SO) Windows, Mac ou Linux e permite importar ficheiros de vetores, geometria e bitmap, como, por exemplo, .dxf, .stl ou .jpg.

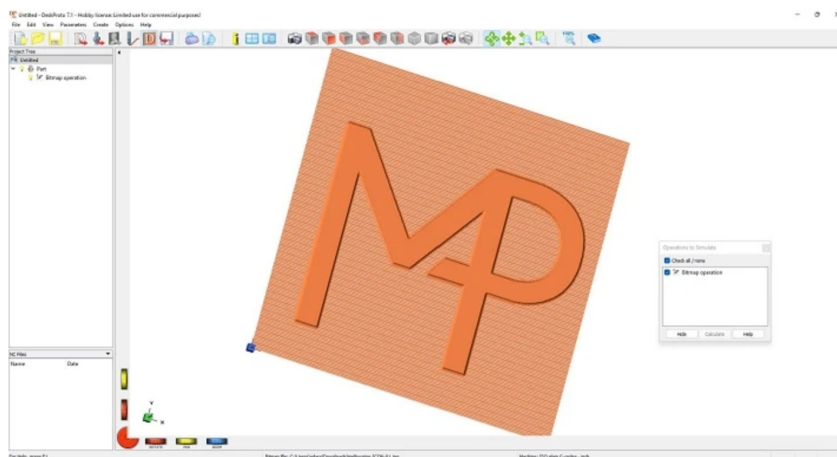


Figura 3.36: Simulação de uma peça maquinada no *software* DeskProto [46]

3.5.4 *Software* e *firmware* para controlo de máquinas CNC

Os *softwares* disponíveis para controlar máquinas CNC têm o objetivo de enviar o G-code para a máquina e monitorizar o progresso da mesma em tempo real. Existem diversas opções de *software* no mercado, como **Universal Gcode Sender (UGS)**,

Grbl Controller (Candle), **MACH4**, entre outros. Ao contrário do *firmware*, que está embutido no *hardware* da máquina, estes *softwares* conseguem operar num PC ou dispositivo eletrónico, gerindo as operações da máquina CNC através de interfaces de utilizador.

O **UGS**, ilustrado na **Figura 3.37**, é uma plataforma gratuita de G-code, dedicada à interação com controladores CNC avançados, como o **GRBL**, **TinyG**, **g2core** e **Smoothieware**. Esta aplicação Java é compatível com a maioria dos SO, incluindo o Windows, Mac ou Linux. Este *software* permite a visualização em 3D do G-code, com segmentos de linha codificados por cores e *feedback*, em tempo real, da posição da ferramenta. Além disso, oferece funcionalidades para otimização do G-code, como remoção de comentários e ajuste da precisão decimal [47].

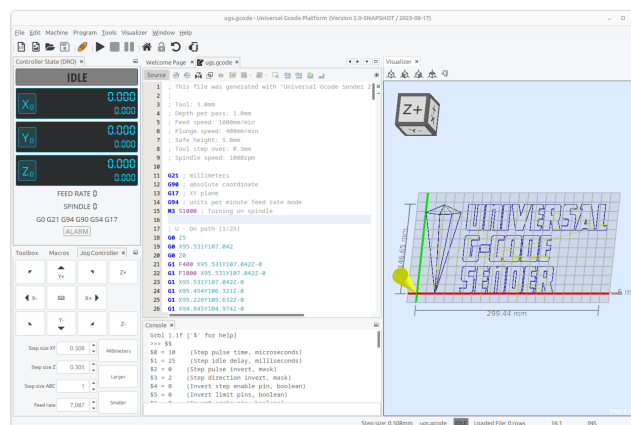


Figura 3.37: Visualização de G-code e maquinagem no *software* UGS [47]

O **Mach4**, é uma versão recente de *software* de controlo de movimento CNC, sucedendo ao amplamente utilizado Mach3. A principal diferença entre os dois é que o Mach4 oferece melhorias significativas na qualidade, velocidade e capacidade de utilizar ou aprender controlos e métodos industriais. Este *software* permite a simulação de diversas máquinas, incluindo fresadoras e tornos, conforme ilustrado na **Figura 3.38**.

Consegue-se perceber que existe uma relação direta entre os *softwares* e *firmwares* das máquinas CNC, pois é necessária haver uma comunicação direta entre ambos. Os *firmwares* utilizados em sistemas CNC são *softwares* embutidos no *hardware* do controlador CNC, tendo elevada importâncias em vários aspetos:

- Fundamental na interpretação do G-code recebido, ao ler as instruções destes comandos, convertendo-os em sinais elétricos que vão controlar os motores da máquina;

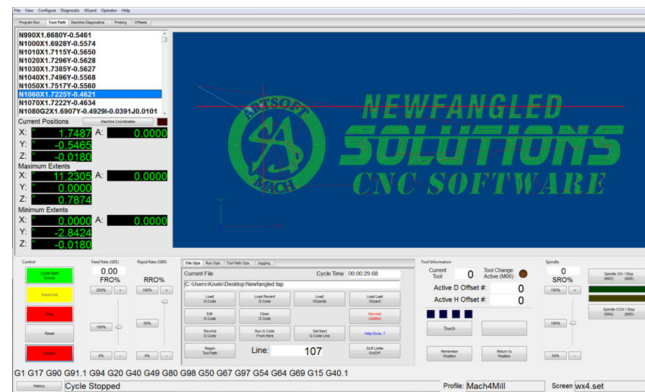


Figura 3.38: Software MACH4 [48]

- Gere os movimentos dos eixos, em sincronia, para a ferramenta de corte seguir os *paths* planejados, com precisão, controlando a aceleração e desaceleração e as velocidades de funcionamento;
- Controla as interfaces I/O tais como os sensores, interruptores de fim de curso e botões de emergência;
- Podem fornecer informação em tempo real da posição dos eixos, do estado de operação da máquina, para serem detetados possíveis erros de operação da máquina [49] [50].

Existem várias opções de *firmware* disponíveis, no mercado, para utilização nas máquinas CNC, sendo os mais comuns o **Marlin**, **Grbl**, **Smoothieware**, entre outros.

O **Marlin** é o *firmware open source* mais conhecido pelos fabricantes de máquinas CNC e impressora 3D. Escrito em C++, foi originalmente, desenvolvido para impressoras 3D, ele também pode ser configurado para controlar máquinas CNC, devido ao seu suporte para uma ampla gama de *hardware*. Inicialmente, foi programado com base em *firmwares* como o Grbl e o Sprinter. É bastante versátil e pode ser adaptado a diferentes sistemas CNC, como fresadoras e torneadores ou até mesmo máquinas de corte a laser.

Uma das principais razões para este *firmware* ser amplamente utilizado é a sua base na plataforma Arduino, permitindo a sua operação numa grande variedade de microcontroladores de baixo custo. Esta compatibilidade com o Arduino torna o Marlin acessível para a fabricantes que desejam construir ou modificar as suas próprias máquinas CNC sem incorrer em grandes custos.

O Marlin tenta suportar todas as placas e configurações de máquinas possíveis, sendo, facilmente, configurado, personalizável, extensível e económico. Este *firmware* inclui um sistema de movimento inteligente que antecipa os *paths*, permitindo a ocorrência de interrupções e aceleração linear, no sistema CNC. Além

disso, possibilita a configuração de máquinas CNC de até 10 eixos de coordenadas independentes, lineares ou de rotação, ideal para aplicações personalizadas. Também oferece suporte para a transferência de G-code através de cartão *Secure Digital* (SD) e permite a implementação de uma interface de utilizador através de um menu num ecrã *Liquid Crystal Display* (LCD), facilitando a operação sem a necessidade de um computador ligado continuamente [51].

Outro *firmware* bastante utilizado é o **Grbl**, sendo igualmente *open source* e gratuito, projetado especificamente para controlar o movimento de máquinas CNC. Escrito em C, o Grbl pode ser executado numa ampla variedade de microcontroladores Arduino, sendo extremamente eficiente em termos de recursos, o que o torna ideal para dispositivos com limitações de memória e processamento. Este *firmware* utiliza G-code como entrada e gera o controlo do movimento através do Arduino, oferecendo precisão e confiabilidade na execução das operações de usinagem.

O Grbl destaca-se pela sua simplicidade e por ser otimizado para operar máquinas CNC com alta eficiência, mostrando-se bastante popular entre os fabricantes de pequenas máquinas CNC pela facilidade de configuração e pelo seu excelente desempenho em *hardware* de baixo custo. Com o Grbl, é possível operar variados tipos de máquinas CNC, como torneadoras ou fresadoras, com um desempenho elevado e sem a necessidade de uma quantidade excessiva de recursos computacionais [52].

3.6 Maquinagem de objetos

A maquinagem de objetos consiste na alteração da geometria inicial de um objeto, removendo parte do seu material, com a utilização de ferramentas próprias para o efeito. São várias as operações de maquinagem que podem ser realizadas, como, por exemplo, torneiar, fresar, roscar e perfurar, existindo três tipos de movimento durante o processo: movimento principal ou de corte, movimento de avanço e movimento de penetração.

Como neste documento só são abordadas máquinas CNC de fresar e torneiar, apenas são explicados, com maior detalhe, estes dois processos.

3.6.1 Fresagem

O processo de **fresagem** consiste na remoção de material utilizando uma ferramenta rotativa, a fresa, com aresta múltipla, onde cada aresta levanta uma quantidade de material, por cada rotação do eixo realizada, sendo que a ferramenta se encontra fixa. Assim, é atingida uma boa taxa de remoção de material, grande qualidade do acabamento da superfície fresada, em conjunto com uma boa precisão [27] [53].

Este tipo de maquinagem envolve operações simples de facejamento e contorno, mas, ao mesmo tempo, a maquinagem de uma variedade de formas complexas [54].

Deste modo, estas máquinas, são, normalmente, utilizadas com o intuito de produzir peças com contornos bem definidos e superfícies curvas, através de peças planas, em bruto, de vários tipos de materiais [27] [53].

É possível ocorrer dois tipos diferentes de fresagem, conforme a posição relativa entre a peça e a ferramenta:

- **Fresagem cilíndrica**, exemplificada na **Figura 3.39a**, em que o eixo da fresa está numa posição paralela à superfície de trabalho da peça onde se realiza a operação de corte;
- **Fresagem frontal**, como a **Figura 3.39b** mostra, onde o eixo da fresa se encontra perpendicular à superfície de trabalho [54].

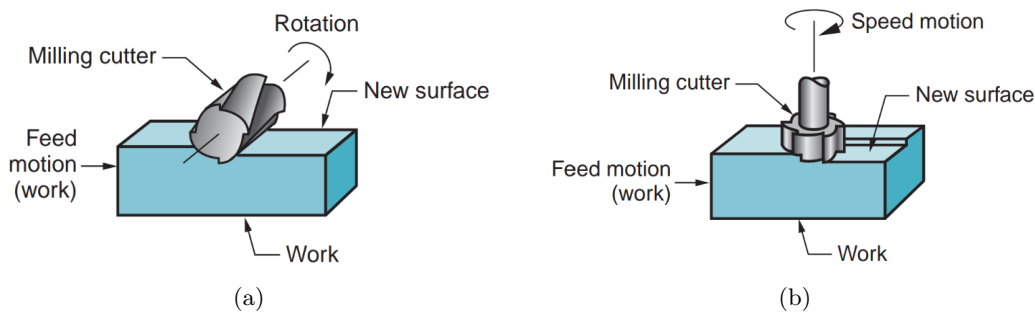


Figura 3.39: Esquema de maquinagem de objetos pelo processo de fresar: cilíndrica (a) e frontal (b) [53]

Quando se fala em movimento relativo entre a peça e a ferramenta, existem dois tipos de fresagem:

- **Fresagem em concordância**, onde o movimento de corte da ferramenta é favorável à direção de avanço da peça. Aqui, quando a aresta de corte contacta a peça, a espessura da apra atinge o seu valor máximo;
- **Fresagem em oposição**, que exhibe um movimento de corte da ferramenta contrário à direção de avanço da peça. Neste tipo de fresagem, a espessura da apra tem um valor mínimo no momento em que a ferramenta entra em contacto com a peça, provocando produção de calor, devido ao atrito inicial da aresta com o material. Este fator pode reduzir a vida útil da ferramenta, bem como o processo de corte. Apesar disso, como o esforço do corte é gradual, este método é o mais utilizado em máquinas que apresentam baixa potência [54].

As operações de fresagem, são, habitualmente, conseguidas com a ferramenta de corte montada num eixo ou, em alternativa, na árvore, sendo que a mesma gira a

velocidades elevadas. A peça de trabalho é fixada à mesa de trabalho da máquina, e é o movimento da mesa que controla o avanço da peça, em relação, à fresa rotativa.

Ao contrário do processo de tornear, a direção do movimento de avanço é perpendicular ao eixo de rotação da ferramenta, e à medida que a peça de trabalho avança os dentes da fresa vão retirar o material da superfície da peça de trabalho, conforme a forma pretendida.

A velocidade do movimento é fornecida pela fresa rotativa, por isso, além da rotação, a fresa não tem qualquer outro tipo de movimento [53].

3.6.2 Torno

O processo de maquinagem por **torno**, também conhecido como processo de **tornear**, utiliza uma máquina-ferramenta chamada torno, com o objetivo principal de se obter peças cilíndricas ou com formas simétricas rotacionais, uma vez que a peça se movimenta em torno do seu próprio eixo [54].

As máquinas do tipo torno mantêm a peça de trabalho entre dois suportes rígidos, de nome mandril ou placa frontal, que permitem que a peça gire. A ferramenta de corte é fixada e suportada no porta-ferramentas, alimentado contra a peça em rotação. Esta ferramenta, é utilizada para remover material à peça de trabalho, que se encontra em rotação, para se obter uma forma cilíndrica, como se pode ser na **Figura 3.40** [27] [53].

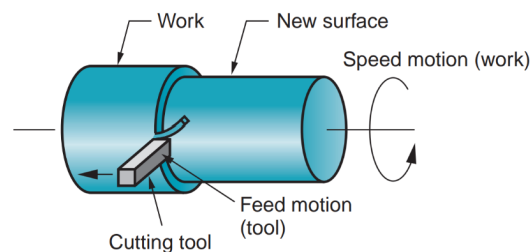


Figura 3.40: Esquema de maquinagem de objetos pelo processo de tornear [53]

Nas operações de torno, o movimento de corte é o movimento de rotação da peça e, normalmente, estas operações são realizadas com a ferramenta de corte alimentada paralelamente ao eixo da peça, em ângulos retos ou, ainda, num ângulo relativo ao eixo da peça [54].

A velocidade com que a peça é torneada depende da rotação da peça de trabalho e expressa-se em milímetros por segundo, quando se trata de velocidade linear ou de corte, e em rotações por segundo, ao se indicar a velocidade angular.

Por outro lado, o movimento de avanço é o movimento longitudinal da ferramenta de corte, relativamente à peça, definindo assim a espessura da aparta, pois o

movimento de avanço obtém-se pela velocidade com que a ferramenta se move relativamente à peça, expressando-se em função do tempo, em milímetros por minuto, ou em função da rotação, em milímetros por rotação [27] [53].

O movimento de penetração produz-se pelo movimento transversal da ferramenta em relação à peça, ou pelo deslocamento da ferramenta durante o movimento de avanço, determinando a altura da apra, ou profundidade de corte, expresso em milímetros [54].

3.7 Conclusão do capítulo 3

Este capítulo forneceu uma visão abrangente sobre os aspetos fundamentais das máquinas CNC, desde a introdução aos conceitos básicos de coordenação e controlo, até à análise detalhada dos componentes que constituem estas máquinas. A pesquisa incluiu os elementos físicos, como sistemas de acionamento e atuadores, mas também os sistemas de controlo e sensores, cruciais para a operação precisa destas máquinas.

Os tópicos incidiram, igualmente, nas aplicações específicas que este tipo de máquinas permite, na área da ortoprotesia, observando-se a importância da precisão e eficiência na produção de moldes positivos e outros dispositivos médicos. Além disso, abordou-se a evolução tecnológica presente no mercado, destacando os controladores, *softwares* CAM, *softwares* e *firmwares* utilizados no controlo destas máquinas, bem como as interfaces de utilizador que facilitam a interação do operador com o sistema.

Por fim, o capítulo estabeleceu uma base sólida para melhorar a compreensão das tecnologias CNC e as suas aplicações práticas, deixando este tema aberto para discussões mais avançadas sobre a otimização e integração destas tecnologias em processos de fabricação específicos. A análise detalhada dos componentes e dos sistemas de controlo CNC destaca a complexidade e a importância de cada um dos componentes constituintes destes sistemas, para ser conseguido um funcionamento harmonioso das máquinas, reforçando a necessidade de contínua inovação e requalificação para manter a competitividade no mercado.

Capítulo 4

Estudo da máquina CNC ”QRC”

Este capítulo aborda o estudo da máquina CNC ”QRC”, encontrada na empresa Padrão Ortopédico, apresentando as suas características e estrutura, o modo de funcionamento, bem como a tecnologia utilizada para o fabrico dos moldes positivos.

4.1 Estrutura e arquitetura do sistema CNC ”QRC”

A máquina CNC estudada, no presente documento, que pode ser observada na **Figura 4.1**, encontra-se na empresa Padrão Ortopédico - Quelhas, Ribeiro e Cardoso, Lda, no Porto, tendo sido desenhada e fabricada por um engenheiro mecânico, que trabalhava na empresa com o intuito de maquinar moldes positivos em PU, maioritariamente, de forma cilíndrica, para o fabrico de dispositivos médicos.

Face ao objetivo para o qual a máquina foi projetada, entende-se que a mesma deve ter, pelo menos, três eixos de movimento, devido ao tipo de moldes que se pretende obter. Assim, a máquina apresentada, funciona como uma fresadora, utilizando a técnica da revolução para remover o material do bloco de matéria-prima, em bruto.

Pela **Figura 4.1**, é possível observar a forma de paralelepípedo, da máquina, bem como a sua verticalidade, que difere da maioria das máquinas CNC encontradas no mercado. Isto deve-se à área de trabalho que se pretende para a maquinagem, aumentando a sua versatilidade e produção de vários tipos de moldes positivos para a área implicada.



Figura 4.1: Máquina CNC "QRC", disponível na empresa Padrão Ortopédico

O sistema implementado é constituído por um mecanismo com dois eixos principais X e Z, sendo que os mesmos representam um plano de deslocamento vertical e horizontal, respetivamente, e um eixo auxiliar A, rotativo, em torno do eixo Z, que faz a peça girar em torno de si própria, na posição vertical.

Como se trata de uma máquina para fresagem de material, o bloco de PU é colocado na mesa de trabalho, giratória, que corresponde ao eixo A, e a remoção de material acontece ao longo do movimento deste eixo, pela ferramenta de corte, a fresa, que se encontra paralela ao eixo Z, sendo o plano da ferramenta de corte. A remoção de material é maior ou menor em profundidade, ou longitudinalmente, conforme as coordenadas fornecidas pelo G-code ao sistema mecânico dos eixos Z ou X, respetivamente.

O estudo e tecnologias implementadas nesta máquina decorreram do levantamento de dados, junto da empresa, para melhor compreensão e entendimento da sua composição. Na **Tabela 4.1**, encontram-se enumerados os vários componentes constituintes da máquina em estudo.

Tabela 4.1: Componentes da máquina "QRC"

Nome	Referência	Corrente	Tensão	Quantidade
Spindle	Spindle CNC 2200 W	10 A	220 V	1
Motor eixo rotativo	Nema 23 Bipolar 3Nm 23HS45-3504S	3,5 A	4,2 V	2
Motor eixo X	Nema 34 Bipolar 3Nm 34HS59-5004S	5 A	5 V	2
Motor eixo Z	Nema 23 Bipolar 3Nm 23HS45-3504S	3,5 A	4,2 V	1
Controlador arduino	MEGA 2560 REV3 A000067	20-50 mA	5 V	1
Cabo alimentação	USB 2.0 Tipo A/Tipo B MITSUI 6896944	-	-	1
Fonte alimentação	S-350-60	5,9 V	60 V DC	3
Driver eixo rotativo	Digital Stepper Drive DM542T	1 - 4,2 A	20-50 V DC	2
Driver eixo X	Digital Stepper Drive DM860I	2,4 - 7,2 A	20-80 V DC	2
Driver eixo Z	Digital Stepper Drive DM542T	1 - 4,2 A	20-50 V DC	1
Fonte tensão spindle	Inversor tensão YL620 A	10 A	110-220 V AC	1
Collet	ER25	-	-	1
Bomba água	HY - 3500 80 W	0,36 A	220 V	N. D.

4.1.1 Estrutura mecânica da máquina CNC "QRC"

A máquina "QRC" foi concebida recorrendo a perfis de alumínio e placas de acrílico, possibilitando a visualização do interior da área de trabalho da máquina, bem como o fácil acesso ao seu interior caso seja necessário realizar algum tipo de manutenção ou reparação da mesma. As suas dimensões máximas são 2 m de altura por 1,4 m de comprimento.

Para melhor análise da sua estrutura, foi criado um modelo 3D, presente na **Figura 4.2**, com a representação de todos os seus componentes, bem como as dimensões reais da máquina.

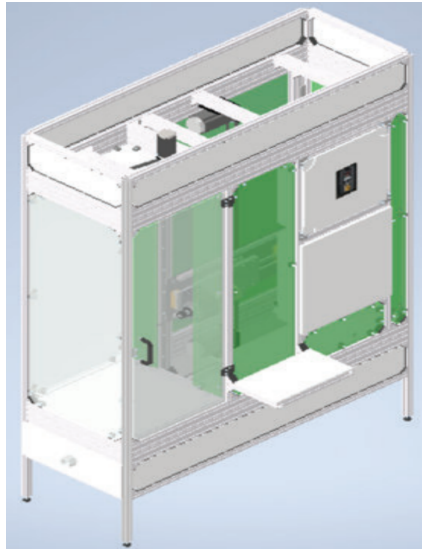


Figura 4.2: Estrutura da máquina "QRC", desenhada através de *software* CAD

Eixos de movimento

Para ser possível ocorrer movimento nos eixos da máquina e para este movimento ser transmitido, em todo o sistema, é necessário existir guias e polias para este efeito. Apesar da máquina em estudo apenas ser constituída por três eixos de movimento, dois lineares e um rotativo, são utilizadas 6 polias, duas correias de transmissão e um fuso de esferas que permitem que o movimento realizado seja o mais correto durante a operação da máquina.

Como referido acima, no **Subcapítulo** 4.1, os moldes obtidos através desta máquina são pela técnica da revolução, que se encontra, igualmente, presente na peça de trabalho a maquinar, na vertical, logo, o percurso vertical da ferramenta de corte é o de maior valor. Por outro lado, o percurso horizontal é de valor inferior, uma vez que a área de trabalho neste plano é menor em comparação ao plano vertical. Além dos percursos referidos dos eixos Z e X, respetivamente, é possível perceber que o eixo que apresenta maior movimento, em qualquer que seja a maquinagem da peça, é o eixo A, o eixo de rotação, sendo possível a colocação de blocos com dimensões até 0,45 m de diâmetro ou 0,45 m de largura e comprimento.

É possível perceber melhor a estrutura dos eixos referidos, na **Figura** 4.3, verificando-se que o eixo X é definido pelo centro da mesa de trabalho, ao longo das guias verticais, com um comprimento de 0,95 m que corresponde ao seu valor máximo. Perpendicularmente a este eixo, está o eixo Z que coincide com o centro da ferramenta de corte, estendendo-se da extremidade da ferramenta, na posição zero, até à sua posição máxima, definido pelo interruptor de limitação implementado, com o total de 0,45 m. Além destes eixos, o eixo de rotação, o eixo A, permite eixos angulares, possibilitando a obtenção de posições num intervalo de 0 a 360°.

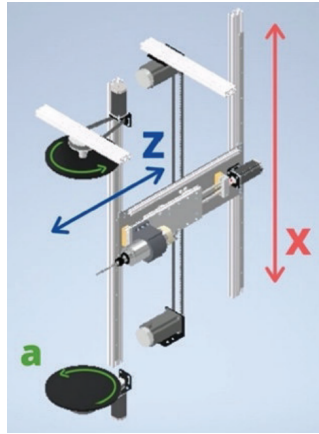


Figura 4.3: Representação dos eixos de movimento X, Z e A, da máquina "QRC"

Pelos valores dimensionais dos eixos da máquina, e pela forma em paralelepípedo da mesma, conseguem-se obter os seguintes valores de área e volume de trabalho, através das **Equações** 4.1 e 4.2, respetivamente:

$$A = C * L = 0,45 * 0,45 = 0,20m^2 \quad (4.1)$$

Onde **C** e **L** são o comprimento e largura máximos, respetivamente, que a ferramenta de corte pode alcançar num plano perpendicular ao eixo de rotação. Esta área representa a secção transversal máxima do bloco que pode ser trabalhada pela máquina.

$$V = A * h = 0,2 * 0,95 = 0,19m^3 \quad (4.2)$$

Onde **h** é a altura máxima de maquinagem da peça. O volume de trabalho define o espaço 3D onde a máquina pode operar, limitando o tamanho máximo dos blocos de PU que podem ser maquinados.

Os valores de área e volume obtidos pelas fórmulas acima apresentadas, são fundamentais para perceber os limites físicos da máquina, bem como as dimensões dos blocos a maquinar e as dimensões em que se pode operar num determinado ciclo de maquinagem. Isto permite planear o processo de fresagem e seleccionar adequadamente os blocos de material a utilizar, otimizando, ainda, a utilização da ferramenta de corte.

O compartimento da área de trabalho para maquinagem da peça, que está representado na **Figura** 4.4, engloba o *spindle*, programado com a velocidade desejada, conforme o material a ser maquinado, e faz com que a ferramenta de corte rode à mesma velocidade que o *spindle* pois encontra-se diretamente acoplada a este, pelo porta-ferramentas.

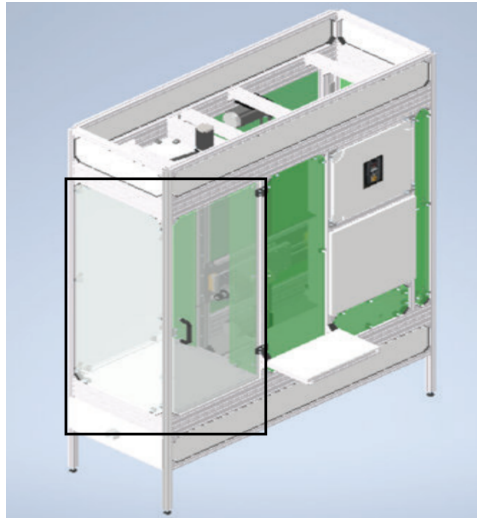


Figura 4.4: Área de trabalho da máquina "QRC"

O eixo de rotação é, então, dividido em dois sistemas de transmissão de movimento que funcionam em simultâneo. A peça de trabalho tanto pode ser encaixada na parte superior como na parte inferior da área de trabalho da máquina. Estes sistemas são explicados seguidamente.

O veio que dá suporte à placa rotativa é fixado à polia do sistema, constituída por 34 dentes e com um passo de 10 mm. O movimento é, então, gerado através da transmissão do eixo A, pela correia dentada, de 65 mm de comprimento e com passo de 10 mm, ligando as duas polias, por tensão. A correia encontra-se ligada ao pinhão de 14 dentes, que está fixado ao veio do motor *National Electrical Manufacturers Association* (NEMA) 23, o qual é um motor de 200 passos, com um ângulo de passo de $1,8^\circ$ (200 passos/revolução), unidade de momento 3 Nm e opera com uma tensão de 4,2 V e corrente de alimentação de 3,5 A.

A par com o movimento descrito, é, igualmente, realizado, um movimento idêntico, no topo da área de trabalho, como se pode ver na **Figura 4.5**. De salientar que, em caso de peças de trabalho de altura superior, poderá haver a necessidade de fixar a peça na parte superior, para evitar oscilações decorrentes da maquinagem.

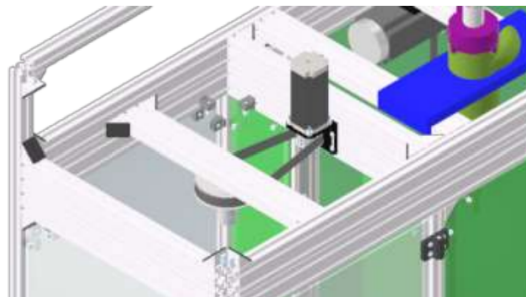


Figura 4.5: Representação do motor e mecanismo de transmissão de movimento relativo ao eixo A, na parte superior da área de trabalho

O movimento vertical da máquina, promovido pelo eixo de movimento X, é constituído por dois motores NEMA 34 que, tal como os NEMA 23. Os motores utilizados para este eixo operam com uma tensão de 5 V e corrente de 5 A, estando ligados por uma correia de 260 mm de comprimento e as suas duas polias, são constituídas por 14 dentes cada e com um passo de 10 mm, estando fixadas por tensão aos dois eixos do motor paralelos. Este mecanismo de transmissão de movimento, do eixo X, e respetivos motores, encontra-se representado na **Figura 4.6**.

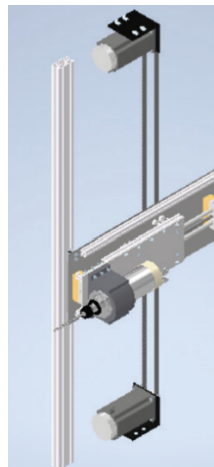


Figura 4.6: Representação dos componentes constituintes do eixo X

O objetivo dos motores, do eixo X, é auxiliar o movimento, quer isto dizer que como o sistema tem de ser movimentado em simultâneo com o *spindle*, os motores auxiliam a que a força seja distribuída por ambos os veios, evitando erros de translação derivados de vibrações excessiva ou força excessiva sofrida pelo eixo, o que pode fazer com que os motores saltem passos.

Para finalizar a descrição dos eixos de movimento da máquina, o eixo Z, maioritariamente responsável pela fresagem da peça de trabalho, por ser mais sensível na sua atuação, utiliza um tipo de transmissão de movimento que difere dos outros eixos mencionados. Assim, o eixo Z, utiliza um mecanismo de transmissão de movimento constituído por um fuso de esferas, de 2 mm de passo e curso de 500 mm, e quatro guias, onde duas são móveis e duas são fixas. Ao fuso de esferas encontra-se acoplado um motor NEMA 23, igual ao do eixo A, que impulsiona a translação horizontal da ferramenta de corte, com a ajuda das guias móveis ao transmitir ao *spindle* o movimento anterior e posterior da ferramenta de corte. Este mecanismo de transmissão de movimento e respetivo motor utilizado encontra-se representado na **Figura 4.7**.

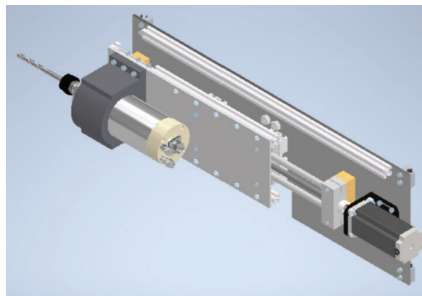


Figura 4.7: Representação dos componentes constituintes do eixo Z

Mesa de trabalho, peça de fixação e ferramenta de corte

Para o bloco de material ser maquinado é necessário que o mesmo seja fixado à **mesa de trabalho**, que neste caso é uma placa rotativa, que se encontra acoplado ao eixo A, representado na **Figura 4.8**. A placa rotativa, que fixa a peça de trabalho ao mecanismo de rotação, encontra-se posicionada num veio paralelo ao veio do motor fazendo com que ocorra a rotação da peça. O movimento de rotação é transmitido por um sistema de duas polias e correia, com o intuito de diminuir a transmissão de potência.

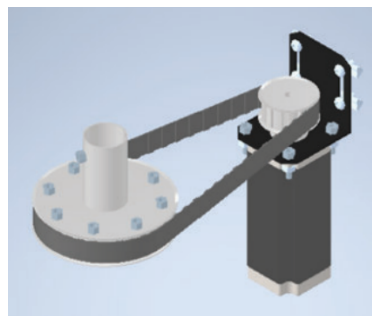


Figura 4.8: Representação do motor e mecanismo de transmissão de movimento relativo ao eixo A

À placa rotativa da mesa de trabalho é então acoplado uma **peça de fixação**, para fixar o bloco a maquinar. Esta peça foi desenhada em 3D, num *software* CAM e impresso numa impressora 3D, tendo um diâmetro de fixação de 180 mm, sendo possível analisar o seu desenho e ficha técnica nas **Figuras A.1 e A.2 do Apêndice A**. Além disso, foi utilizado um tubo com 30 mm de diâmetro para fazer a ligação entre esta peça de fixação e a placa rotativa da mesa de trabalho. Na **Figura 4.9** é possível observar-se a peça referida.

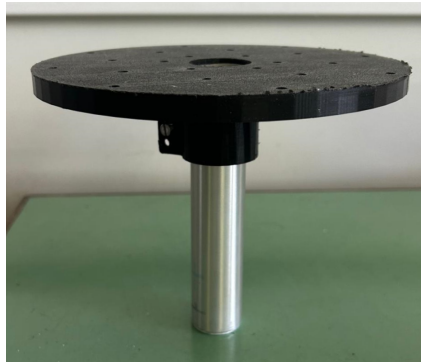


Figura 4.9: Peça de fixação do bloco responsável pela ligação à placa rotativa

Para completar a parte mecânica do processo de maquinagem é necessário referir a **ferramenta de corte** utilizada para o efeito. Este componente é fundamental para a maquinagem das peças ao remover material em excesso do bloco de PU, até ser obtido o molde desejado. Na **Figura 4.10** encontra-se exemplificado a ferramenta de corte utilizada nesta máquina.



Figura 4.10: Exemplo da ferramenta de corte utilizada na máquina "QRC" [55]

A ferramenta, tem um diâmetro de 10 mm, ponta esférica e 4 flautas de corte e é concebida em aço de alta velocidade, permitindo a resistência ao desgaste e ao calor gerado durante o processo de corte.

4.1.2 Sistema de transmissão de movimento

No **Subcapítulo 4.1**, através da **Tabela 4.1**, abordaram-se sumariamente os motores que fornecem movimento aos mecanismos de transmissão de movimento, utilizados na máquina. Em seguida, são detalhadas as suas características.

Motores utilizados

Como referido, os motores são fundamentais para acontecer o controlo e transmissão de movimento, neste caso para os eixos do sistema da máquina CNC. No caso da máquina em estudo, estão presentes cinco motores, do tipo NEMA, três do modelo 23 e dois do modelo 34, como se pode Observar na **Figura 4.11**, que controlam os eixos da máquina CNC. Os tamanhos dos motores de passo NEMA dependem do

tamanho da carcaça do motor de passo. Nas **Figuras B.1** e **B.2** do **Apêndice B.1** é possível verificar as dimensões dos motores utilizados neste sistema.

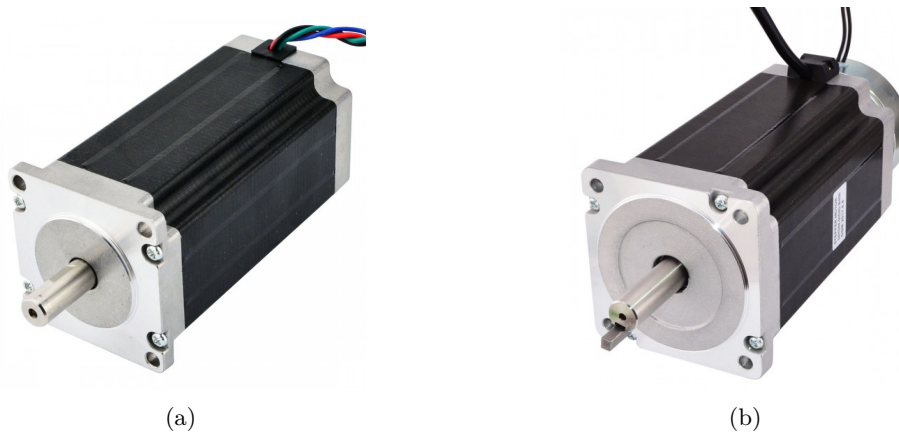


Figura 4.11: Exemplos de motores NEMA 23 (a) e NEMA 34 (b) utilizados na máquina [56] [57]

Os NEMA 23, são motores de passo, bipolares e híbridos, com elevado torque com cerca de 3 Nm, e com dimensões de 58,4*58,4 mm². Estes motores têm um ângulo de passo de 1,8° significando que a cada revolução executa 200 passos e a cada passo realiza 1,8°. Podem ser controlados por duas pontes H, mas recomenda-se a utilização de um *driver* de motor de passo. No projeto da máquina, como referido, são utilizados dois motores deste tipo, nas placas rotativas inferior e superior, apresentando um papel importante no controlo do ângulo e direção exatos de rotação do bloco em torno do seu eixo, como se pode verificar na **Figura 4.12**.

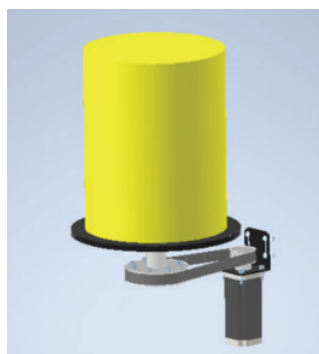


Figura 4.12: Motor NEMA 23, acoplado na parte inferior do prato de rotação da máquina "QRC"

Já o terceiro motor NEMA 23, encontra-se localizado na placa de montagem do *spindle* sendo responsável pelo seu avanço ou recuo. Pela maneira como se apresenta acoplado ao *spindle*, transmite o seu torque, pelo avanço conseguido pelas guias deslizantes da ferramenta de corte, como se pode verificar na **Figura 4.7**, do **Subcapítulo 4.1.1** [56].

Por outro lado, os NEMA 34 utilizados na máquina "QRC", são idênticos aos NEMA 23, diferindo em alguns aspetos: dimensões, com 86,4*86,4 mm²; torque, com cerca de 13 Nm; características de funcionamento, por combinarem o motor de passo de relutância variável com o motor de passo com imã permanente [57]; e por utilizarem taxas de avanço mais elevadas. Estas diferenças permitem que o porta-ferramentas seja transportado verticalmente, apesar do seu peso e, permitem, igualmente, minimizar a vibração e ruído sentidos aquando da maquinagem de peças. A **Figura 4.6**, do **Subcapítulo 4.1.1**, mostra como os dois motores deste tipo, se encontram posicionados na máquina.

Spindle elétrico

Para concluir a descrição dos componentes do sistema de transmissão, outro componente fundamental à maquinagem é o *spindle*, que se encontra no compartimento definido para a maquinagem da peça. Com uma potência de 2200 W, tensão de 220 V e corrente de 10 A, pode ser programado para rodar à velocidade pretendida conforme o material que se pretende maquinar, promovendo, ainda, o movimento de rotação da ferramenta de corte, diretamente acoplada ao mesmo.



Figura 4.13: *Spindle* elétrico encontrado na máquina "QRC" [58]

O *spindle* utilizado nesta máquina encontra-se separado da parte elétrica máquina e é controlado manualmente. Assim, neste caso, e apesar de ser um componente importante para a definição de velocidade de corte, não é necessário serem ajustados estes parâmetros aquando de cada processo de maquinagem. A velocidade com que o *spindle* opera não é definida diretamente, e imposta por uma gama de frequência que varia entre os 85 e 90 Hz, velocidade adequada para maquinar o PU.

4.1.3 Sistema de controlo

Anexado ao compartimento da área de trabalho da máquina, encontra-se a área onde se encontram colocados os componentes eletrónicos e mecânicos que vão controlar a máquina CNC, como se pode observar na **Figura 4.14**.

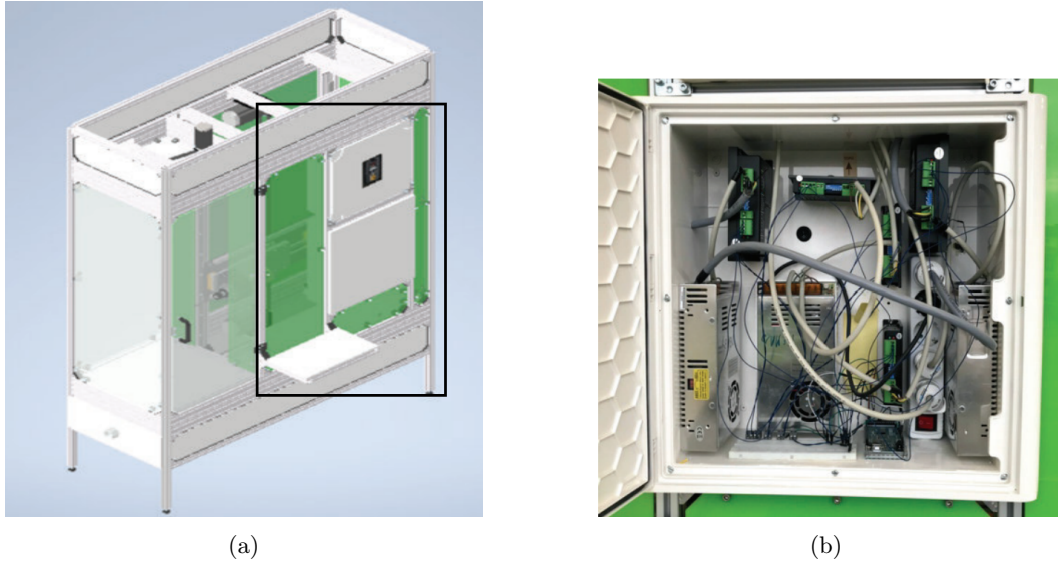


Figura 4.14: Área de controlo (a) e componentes eletrónicos (b), da máquina "QRC"

É nesta unidade, que ocorre a comunicação entre o operador e a máquina, uma vez que é aqui que é inserido o G-code, no controlador, enviando até aos terminais do motor, os comandos necessários para que o funcionamento da máquina seja o mais síncrono e correto possível.

Como referido, esta área é a responsável pelo controlo da máquina e é aqui que se encontram os *drivers* dos motores bem como o controlador. Além disso, também é aqui que são realizadas as comunicações entre a máquina e o operador pela da ligação do controlador a um PC.

Acima da área de controlo, onde se encontram as ligações elétricas, encontra-se um controlador manual do *spindle* para caso seja necessário, o mesmo poder ser acionado manualmente, em emergência ou mal funcionamento da parte elétrica e eletrónica. Este componente é abordado no **Subcapítulo 4.1.3**.

Microprocessador Arduino Mega 2560 Rev3

O microprocessador escolhido para o projeto é o Arduino Mega 2560 REV3, igual ao da **Figura 4.15**, um *microcontroller unit* (MCU) com base no ATmega2560. Com 256 KB de memória *Flash*, 8KB de *Static Random Access Memory* (SRAM) e 4 KB

de *Electrically Erasable Programmable Read-Only Memory* (EEPROM), pode suportar programas complexos e armazenar dados de modo persistente, ideal para guardar dados mesmo quando o dispositivo é desligado. Ainda, devido à sua configuração de I/O digitais e analógicos, bem como as ligações permitidas com *hardware* torna-se um microprocessador bastante completo, pela sua versatilidade de capacidade. No **Apêndice B.3** encontra-se o esquema de ligações possíveis a este microprocessador.

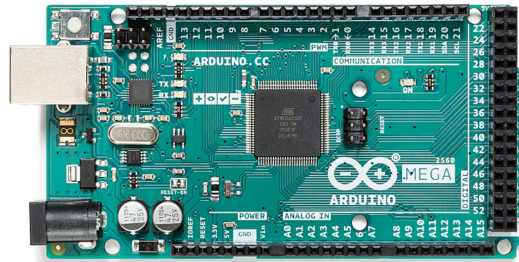


Figura 4.15: Microprocessador Arduino Mega 2560 REV3 [59]

O MCU presente neste componente, ATmega2560, opera com uma tensão de 5 V e a sua alimentação recomendada pode ser feita por uma entrada de 7 a 12 V, ou no limite de 6 a 20 V. Com uma frequência de clock de 16 MHz, pode realizar operações de modo fácil e eficiente, o que é adequado para uma ampla variedade de aplicações.

É neste MCU que se encontra inserido o *firmware* escolhido, utilizado para interpretar o G-code e o controlar a máquina CNC, sendo responsável pelo pós-processamento na maquinagem das peças.

Mais adiante, no **Subcapítulo 4.1.3**, são abordados o *firmware* e controlador utilizados neste projeto.

Drivers dos motores

Quando são utilizados motores de passo, regra geral, é necessário serem utilizados *drivers* que vão controlar estes motores, convertendo os sinais provenientes do controlador em tensão e corrente adequadas para os mesmos serem acionados, proporcionando movimentos precisos, suaves e eficientes. A par com isto, são estes componentes que protegem os motores da sobrecarga de tensão e corrente.

No presente projeto, são utilizados dois tipos de *drivers* de passo digitais, como se pode observar na **Figura 4.16**, dois DM860I e três DM542T, associados aos respetivos motores NEMA 34 e NEMA 23.

O *driver* DM860I, com uma tensão e frequência de entrada de 20 a 80 V DC e 200 kHz, respetivamente, e com a possibilidade de escolha de até oito correntes de pico de 2,4 a 7,2 A, é adequado para motores de duas e quatro fases. Permite uma



Figura 4.16: *Drivers* utilizados na máquina "QRC": DM860I (a) e DM542T (b) [60] [61]

configuração rápida para a utilização dos motores devido à sua função de autoidentificação e autoconfiguração dos parâmetros. Devido ao seu torque otimizado, elimina a instabilidade em rotações médias, oferece menor ruído e aquecimento ao operar. Além disso, aquando da sua iniciação, promove um início de movimento suave, sem "saltos", e permite a seleção de até dezasseis micropassos de resolução [60].

Em contraste, o *driver* DM542T é bastante idêntico em funcionamento, quando comparado com o DM860I, sendo as diferenças encontradas nos valores de tensão e corrente, podendo operar entre 18 e 50 V DC e entre oito correntes de pico de 1 a 4,5 A.

Nas **Figuras B.7** e **B.8** do **Apêndice B.7** é possível serem observadas as dimensões de ambos os *drivers* referidos [61].

Fonte tensão do *spindle*

A fonte de tensão *spindle* é utilizada para controlar a velocidade e o torque dos motores, ajustando a frequência e a tensão de entrada fornecidas aos mesmos. A utilização deste tipo de componente, pode melhorar a eficiência energética, ao permitir ajustar a operação dos motores às necessidades da aplicação, evitando o consumo excessivo de energia.

Uma vez que neste estudo o *spindle* não se encontra ligado ao controlador, é necessário utilizar uma interface manual, para o mesmo ser controlado e, assim ajustar os seus valores de operação.

Deste modo, é utilizado o inversor de frequência, YL620-A, ou conversor de frequência, como se visualiza na **Figura 4.17**, que permite o controlo do torque e modos de operação automáticos e manuais dos motores. Além disso, oferece proteção ao motor evitando a sobrecarga, curto-circuito, sobretensão e sobreaquecimento, o que aumenta a segurança e a vida útil do mesmo. Através da **Figura B.3** do

Apêndice B.2 é possível observar as ligações elétricas disponíveis deste componente [62].



Figura 4.17: Inversor do *spindle* YL620-A [62]

Fonte de alimentação

Como referido no **Subcapítulo** 3.2.6, as fontes de alimentação são componentes cruciais para o funcionamento das máquinas CNC, responsáveis por fornecer a energia necessária para o funcionamento eficiente dos motores de passo, servomotores e outros sistemas eletrônicos. Uma fonte de alimentação de qualidade garante que a máquina opere com precisão e estabilidade, prevenindo falhas e garantindo a segurança dos componentes, protegendo-as de sobrecargas, sobretensão e sobreaquecimento.

Nesta máquina, são utilizadas três fontes de alimentação, iguais às da **Figura** 4.18, adequadas a motores de passo, com dimensão de 215*115*50 mm³ cada, entrada de tensão ajustável, com um interruptor, entre os 115 e 230 V AC e saída de 60 V DC e 5,9 A. Adicionalmente, tem embutido uma ventoinha DC que provoca o arrefecimento do sistema, o que previne o sobreaquecimento do sistema. Ainda, tem uma baixa ondulação de saída e ruído e protecção contra sobrecarga, sobretensão e curto-circuito [63]. Estas características, em conjunto com o seu baixo custo, tornam esta fonte de alimentação eficiente e acessível.

Na **Figura** B.5 conseguem-se analisar as dimensões das fontes de alimentação utilizadas neste projeto.

Firmware e controlador Grbl

Como mencionado no **Subcapítulo** 3.5.4, o *software* e *firmware* para controlo da máquina CNC são de grande relevância, pois enviam e interpretam, respetivamente, o G-code, utilizado nos processos de maquinagem, monitorizando-os em tempo real. Além disso, permitem a interação entre o operador e máquina, através do interface do utilizador que pode ser qualquer dispositivo eletrónico ou PC.



Figura 4.18: Fonte de alimentação S-350-60 utilizada na máquina [63]

O *firmware* utilizado na máquina em estudo é o **Grbl**, pelo elevado desempenho e baixo custo de implementação e pela facilidade de utilização com os controladores Arduino existentes no mercado, pois uma vez que na máquina são utilizados componentes com limitações de memória e processamento. Este *firmware* existem várias versões de acordo com as funcionalidades da máquina CNC que se pretende obter. Como a máquina em estudo tem três eixos, dois lineares e um de rotação, é necessário utilizar a versão 5/6 eixos, mais propriamente, Grbl-Mega-5X (**Figura 4.19**), compatível com o MCU utilizado, Arduino Mega 2560 REV3.

grbl-Mega-5X

Figura 4.19: Logótipo da versão 5/6 eixos Grbl-Mega-5X [64]

Este *firmware* consegue manter até 30 kHz de pulsos de controlo estáveis sem ocorrer oscilação e tem uma gestão completa de aceleração, conseguindo prever até 24 movimentos, planeando as velocidades impondo acelerações suaves e curvas sem "saltos". Além destas particularidades, aceita G-code compatível com os padrões e podem ser utilizados várias extensões de ficheiros CAM, permitindo variados movimentos, como arcos, círculos e helicoidais [64].

No **Apêndice C.1** é possível visualizar-se o G-code compatível com o Arduino Mega 2560.

De outro modo, o *software* utilizado para complementar o Grbl é o **Grlb Controller**, que tem como principal objetivo ser o canal de comunicação entre o *firmware* e o operador. Ou seja, ele permite que o operador envie o G-code diretamente para a máquina CNC, além de monitorizar o progresso da maquinagem em tempo real. Para ser simplificado este processo, é utilizada a interface gráfica do *software*, exemplificada na **Figura 4.20** facilitando a visualização das coordenadas, a definição de parâmetros de corte, e o ajuste de posições, tornando o processo mais eficiente e intuitivo.

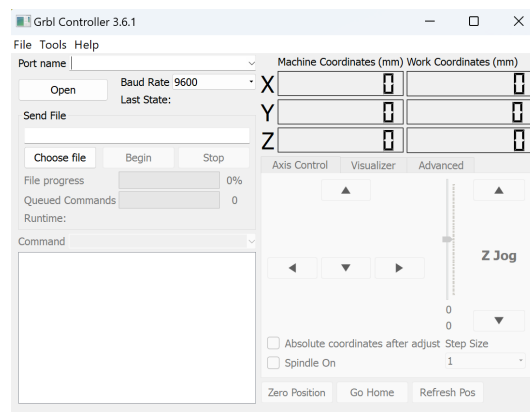


Figura 4.20: Página de arranque do *software* Grbl Controller

O Grbl Controller também oferece ao operador a possibilidade de enviar comandos manuais diretamente através de uma linha de comando, utilizando códigos próprios do Grbl que podem ser observados no **Apêndice C.2**. Esses comandos são especialmente úteis para configurar a máquina antes de iniciar um ciclo de maquiagem, ajustar parâmetros em tempo real ou corrigir erros durante o processo de maquiagem.

Este *software* também oferece variadas funções automáticas que facilitam a maquiagem, como a definição automática de referências para os eixos e a simulação do percurso da ferramenta, permitindo ao operador prever o tempo estimado de maquiagem e identificar potenciais problemas antes de iniciar o processo.

Ainda, aquando da execução do G-code, o Grbl Controller mostra em tempo real a posição da ferramenta, o estado do *spindle*, e os valores de velocidade e avanço, permitindo ajustes dinâmicos e uma visão geral do desempenho da máquina.

4.2 Funcionamento da máquina CNC "QRC"

Para uma melhor compreensão do funcionamento da máquina CNC "QRC", este **Subcapítulo** apresenta um resumo da interação entre o *software*, o *firmware* e o *hardware* da máquina. São descritos como os movimentos dos eixos e do *spindle* são controlados através desta interação.

Além disso, também é abordada a interação entre o operador e máquina, através da MMI, que facilita a configuração e monitorização da maquiagem em tempo real. Esta análise é fundamental para entender a eficiência e o desempenho da máquina no fabrico de moldes positivos.

A máquina pode ser dividida em diferentes sistemas, como a estrutura mecânica, o sistema de transmissão de movimento, o controlador e a MMI, conforme ilustrado na **Figura 4.21**. Este diagrama destaca as ligações entre o *software*, *firmware*, controlador e os sistemas mecânicos responsáveis pela execução da maquiagem.

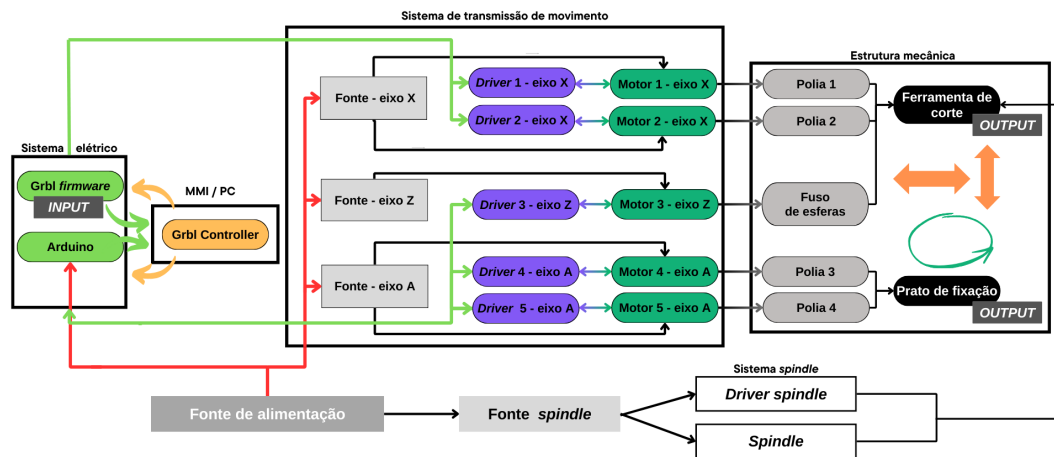


Figura 4.21: Diagrama ilustrativo das ligações estabelecidas entre o *software*, *firmware* e controlador e os sistemas de transmissão de movimento e mecânicos

Na **Figura 4.21** acima apresentada, consegue-se verificar a interação que acontece entre o *software*, *firmware* e controlador da máquina, bem como o envio da informação pelo *software* para os *drivers* dos motores e posteriormente para os motores que vão transmitir movimento aos componentes mecânicos: polias e fuso de esferas e ferramenta de corte e prato de fixação, sucessivamente.

Seguidamente, é detalhado como é efetuada esta interação para ser obtida a peça final maquinada.

4.2.1 Controlador, *software* e *firmware*

O controlador de uma máquina é o *core* do sistema sendo responsável pela interpretação dos comandos e pelo controlo preciso dos movimentos dos eixos e do *spindle*. No caso desta máquina, como referido no **Subcapítulo 4.1.3**, o controlador utilizado é baseado no Arduino Mega 2560, que trabalha em conjunto com o *firmware* Grbl para garantir o controlo da mesma.

A utilização do Grbl, nomeadamente o Grbl-Mega-5x, deve-se à sua compatibilidade com o controlador Arduino e pela capacidade em gerir, eficientemente, até cinco eixos de movimento, permitindo que a "QRC" opere com os eixos X, Z, e A. Este *firmware* é responsável por interpretar o G-code, que define os movimentos e operações da máquina durante a maquinação.

O processo de maquinação da peça só é iniciado quando o G-code é inserido no Grbl Controller, que funciona como uma interface entre o operador e a máquina. Como referido no **Subcapítulo 3.1.1**, o G-code é um conjunto de instruções que a máquina CNC utiliza para realizar movimentos precisos. Como se observa **Tabela 3.1**, do **Subcapítulo 3.1.1**, cada comando controla um aspeto específico da operação

da máquina, essencialmente, comandos de posicionamento, de controlo do *spindle* e de movimento circular:

- Os comandos de posicionamento, como G00 e G01, movem a ferramenta de corte numa trajetória linear, em que G01 é utilizado quando a ferramenta fresa o material;
- O comando M03, ativa a rotação do *spindle*, enquanto o M05 desliga a rotação, comandos de controlo do *spindle*. Com estes comandos é possível ajustar a velocidade do *spindle* conforme os parâmetros definidos;
- Os comandos de movimento circular, como o G02 e G03 são utilizados para gerar movimentos circulares, no sentido horário e anti-horário, respetivamente. Estes comandos são cruciais para operações complexas, onde é necessário realizar curvas suaves.

Os comandos são enviados sequencialmente, de forma específica, conforme o programado em G-code e o Grbl garante que cada um seja executado com precisão, otimizando a aceleração e desaceleração dos motores para evitar movimentos bruscos que possam comprometer a precisão da maquinagem.

Assim que o G-code é carregado, o *firmware* Grbl processa as suas instruções armazena-as temporariamente no *buffer* do controlador. Depois de recebida a informação no *buffer* do controlador, o *firmware* guarda-a e trata-a convertendo-a em sinais elétricos, que são enviados para os *drivers* de cada motor de passo. A interpretação desses comandos pelo Grbl garante que o movimento ocorra de forma suave e precisa.

Esses *drivers* são responsáveis por controlar os motores que executam os movimentos dos eixos X, Z e A, bem como a rotação do *spindle*. Cada motor de passo é ativado em sequência, com base nos comandos que o *firmware* processa, permitindo que a ferramenta de corte siga precisamente o percurso definido. Esta transmissão de movimento pode ser observada na **Figura 4.21**, onde a ferramenta é o *output* do sistema, realizando movimentos verticais ou horizontais, exemplificados pelas setas laranja, ou rotativos, exemplificados pelo movimento rotação a verde, no prato de fixação do bloco. Este processo de interpretação e conversão de sinais é fundamental para garantir que a operação de maquinagem seja executada com precisão.

Durante o processo de maquinagem, o Grbl também realiza cálculos para otimizar as acelerações e desacelerações dos eixos, garantindo movimentos suaves que minimizam o risco de erros que podem comprometer a precisão da maquinagem.

Este ciclo de execução é repetido continuamente até que todos os comandos no G-code sejam processados e o molde positivo seja completamente maquinado. A sincronização precisa entre o *software*, *firmware* e *hardware* assegura que o molde seja produzido com as características exatas definidas no projeto.

4.2.2 *Man Machine Interface* da máquina CNC "QRC"

A MMI da máquina CNC "QRC" desempenha um papel crucial na interação entre o operador e a máquina, permitindo a configuração dos parâmetros de maquinagem, controlo dos movimentos da máquina e monitorização do progresso de maquinagem em tempo real, para garantir que o processo acontece eficientemente e sem ocorrências.

Como referido anteriormente e como se observa acima na **Figura 4.21**, a MMI é o *input* da informação enviada para a máquina, que neste caso acontece através de um PC através do Grbl. Quer dizer que a sua implementação é conseguida pela combinação de *software* e *hardware*.

Para a MMI ser utilizada no PC é necessário o mesmo ser ligado ao controlador através de um cabo USB, e é aqui que é executado o Grbl Controller que carrega o G-code, definir a posição inicial da ferramenta de corte, ajustar parâmetros de maquinagem, como a velocidade de corte, e iniciar ou pausar o ciclo de maquinagem.

O Grbl Controller permite que o operador visualize as coordenadas dos eixos em tempo real e verifique o progresso da operação de maquinagem. A interface gráfica mostra a posição atual da ferramenta em cada eixo, bem como a velocidade do *spindle* e os parâmetros de corte, permitindo ao operador acompanhar o progresso da maquinagem.

Em caso de erros, como uma sobrecarga do motor ou um desvio inesperado da ferramenta, o Grbl Controller emite alertas visuais ou sonoros para avisar o operador de que há um problema. A sobrecarga do motor, por exemplo, pode ocorrer se o *spindle* estiver a girar muito rápido ou se a ferramenta estiver a remover material mais rápido que a capacidade de funcionamento do motor. Um desvio de ferramenta pode acontecer se a peça de trabalho não estiver corretamente fixada ou se a trajetória da ferramenta não for seguida de forma precisa devido a uma falha no sistema de transmissão de movimento.

Quando tais problemas são detetados, o operador pode pausar o processo de maquinagem imediatamente, ajustar os parâmetros conforme necessário (como reduzir a velocidade de avanço ou alterar a posição inicial da ferramenta) e, em seguida, retomar a operação. O *feedback* em tempo real fornecido pelo Grbl Controller reduz o risco de ocorrências na peça ou na máquina, permitindo uma intervenção rápida e eficaz, crucial para garantir a qualidade do molde final.

Além disso, o sistema permite registar os dados da operação, possibilitando uma análise posterior para identificar a causa de eventuais problemas e evitar que se repitam em operações futuras.

Outra característica importante da MMI é a capacidade de definir os "pontos zero" dos eixos com precisão, garantindo que a ferramenta de corte começa na posição correta em relação ao bloco a maquinar. Este aspeto é conseguido pela utilização

dos comandos manuais inseridos diretamente no Grbl Controller, como o ajuste do eixo Z para o centro do eixo A.

4.3 Conclusão do Capítulo 4

Neste capítulo, foi analisada a estrutura, funcionamento e desempenho da máquina CNC "QRC", utilizada para o fabrico de moldes positivos em PU.

Em primeiro lugar, foi realizada uma análise detalhada à estrutura mecânica da máquina, destacando os três eixos principais e o papel fundamental de cada um no processo de maquinagem.

Seguidamente, foi descrito o sistema de controlo, que envolve a interação do *software*, *firmware* e controlador Arduino para coordenar os movimentos da máquina de maneira precisa.

Também foi abordado o processo de fabrico de moldes positivos, desde a digitalização do modelo até ser gerado o G-code, com ênfase na preparação do material e na execução dos ciclos de maquinagem. Foram identificadas as duas principais fases do processo de fresagem: a fresagem inicial e o acabamento, ambos essenciais para garantir a qualidade e a precisão do molde final.

Por fim, foram apresentadas algumas limitações encontradas na máquina, tenta encontrar sumariamente alternativas para a sua melhoria.

Capítulo 5

Desenvolvimento do projeto

Este capítulo descreve o processo de desenvolvimento do projeto, focando a análise das ligações elétricas da máquina CNC "QRC" e na verificação da integração entre o controlador e o computador para permitir a maquinagem. O objetivo é detalhar a integração do *hardware* e *software*, assegurando o controlo eficiente da máquina.

É realizada uma verificação das ligações e componentes, garantindo a compatibilidade com o *firmware* Grbl e discute-se a ligação da máquina ao PC através de uma interface gráfica para enviar comandos e monitorizar a operação de maquinagem. As configurações dos pinos, ajustes do Grbl, e o controlo do *spindle* são também abordados, com as soluções para as dificuldades encontradas.

Ainda é exemplificado o processo que envolve a obtenção do molde positivo maquinado, desde a digitalização do membro até à configuração da máquina através de *software* CAM e a maquinagem do bloco de PU. Por fim, são apresentados os testes de maquinagem, bem como os resultados obtidos com a mesma, para avaliar a precisão e desempenho da máquina.

5.1 Ligações elétricas do *hardware* do sistema

Para perceber se a máquina CNC em estudo se encontrava a funcionar adequadamente foi importante fazer um estudo das ligações existentes entre os vários componentes da mesma, os quais foram mencionados no **Capítulo 4**. Após análise das várias ligações estabelecidas, percebeu-se existirem algumas fragilizadas ou mal conectadas, pelo que foi pertinente analisar uma a uma. Assim, são detalhadas as

ligações fundamentais entre os componentes de *hardware* que constituem o sistema de controlo desta máquina.

Como referido anteriormente, no **Subcapítulo 4.1**, o Arduino Mega 2560 desempenha um papel importante na gestão dos sinais de controlo enviados para os motores de passo NEMA 23 e NEMA 34, responsáveis pelos movimentos dos eixos da máquina. Para o funcionamento dos motores ser o mais correto, os sinais enviados pelo Arduino são amplificados pelos *drivers* de motores, nomeadamente os DM542T para os NEMA 23 e os DM860I para os NEMA 34. Estes *drivers* convertem os sinais de controlo em pulsos de corrente adequados para alimentar os motores, permitindo que estes realizem movimentos precisos.

A alimentação do sistema é fornecida pelas fontes de alimentação S-350-60, que garantem a voltagem e corrente adequadas aos *drivers* e respetivos motores. As ligações entre esses componentes são essenciais para o controlo adequado da CNC, sendo cruciais para o desempenho e precisão da máquina durante o processo de maquinagem.

Na **Figura 5.1** encontra-se demonstrada a arquitetura do sistema CNC "QRC" que se encontra estudo e análise.

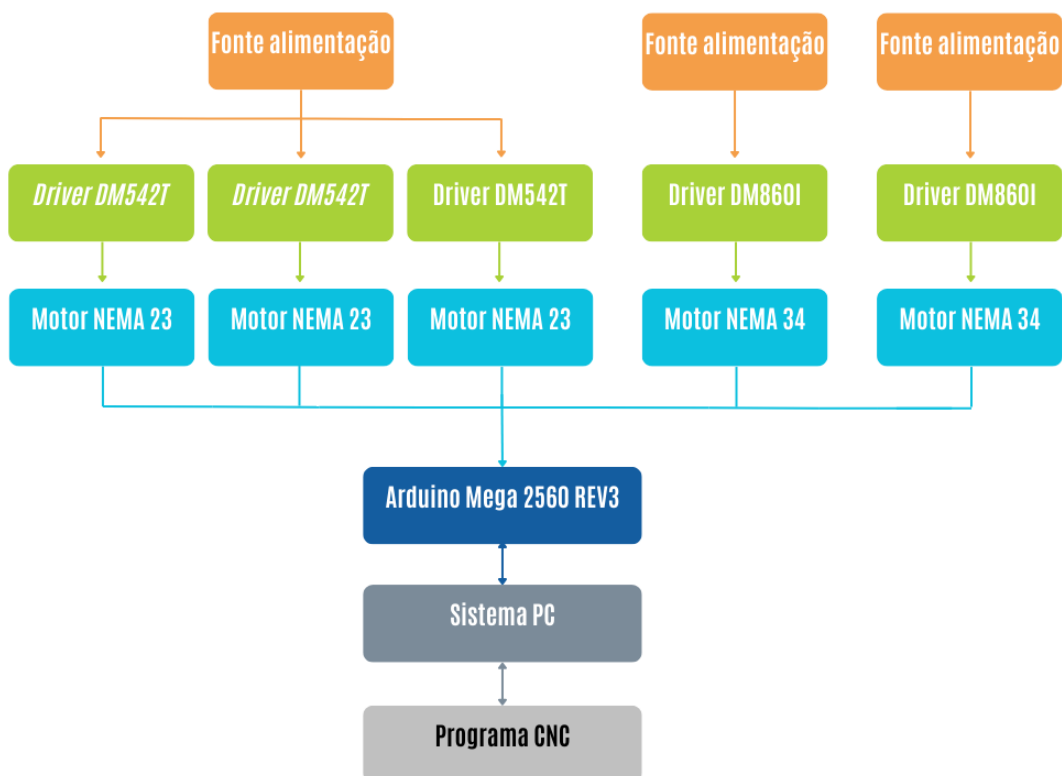


Figura 5.1: Diagrama de blocos da arquitetura do sistema CNC "QRC"

Em seguida, são detalhadas as ligações referidas, frisando a importância de cada componente e a sua interligação.

5.1.1 Ligação entre os *drivers* e os motores

Cada motor de passo utilizado na máquina tem quatro fios de ligação, que correspondem às suas bobinas, e têm o objetivo de realizar a ligação dos motores aos respectivos *drivers*.

Como referido anteriormente, no **Subcapítulo** 4.1.3, os *drivers* utilizados para os motores são os DM542T, para o NEMA 23 e os DM860I para os NEMA 34, sendo responsáveis por controlar a corrente que passa para os motores, pelos sinais que recebem do Arduino, fornecendo potência suficiente para os fazer girar.

Antes de serem mencionadas as ligações dos *drivers* às respectivas fontes de alimentação, conforme referido no **Subcapítulo** seguinte, 5.1.2, é necessário serem analisadas as ligações dos *drivers* aos motores, sendo as mesmas detalhadas a seguir.

Ligação dos motores NEMA 23 aos *drivers* DM542T

Os motores NEMA 23 têm dois pares de bobinas, que correspondem às fases do motor, onde o primeiro par, A e C, está representado a preto e verde, e o segundo par, B e D, está exibido a vermelho e azul, como se pode observar no diagrama da **Figura 5.2** [65].

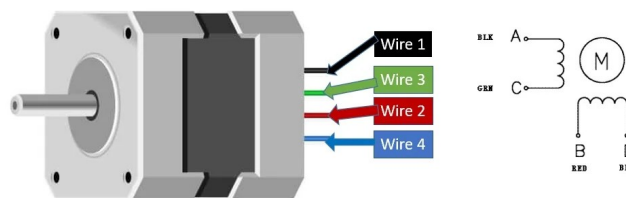


Figura 5.2: Diagrama das ligações possíveis ao motor NEMA 23 [65]

Por outro lado, do lado dos *drivers*, nomeadamente no DM542T, são encontradas vários tipos de ligações, como se observa na **Figura 4.16b** do **Subcapítulo** 4.1.3, destacando-se, nesta etapa, as ligações A+, A-, B+ e B-, responsáveis pelas fases do motor. Assim, a maneira correta de fazer a ligação entre o *driver*, DM542T, e o motor, NEMA 23, é ligar A e C a A+ e A-, respetivamente, e ligar B e D a B+ e B-, respetivamente, como se verifica no esquema da **Figura 5.3** [60].

Para o sistema funcionar corretamente é necessário estabelecer a ligação entre cada um dos motores NEMA 23 e cada um dos respetivos *drivers*. Deste modo, foi necessário serem verificadas as ligações entre cada motor e *driver*, percebendo-se que

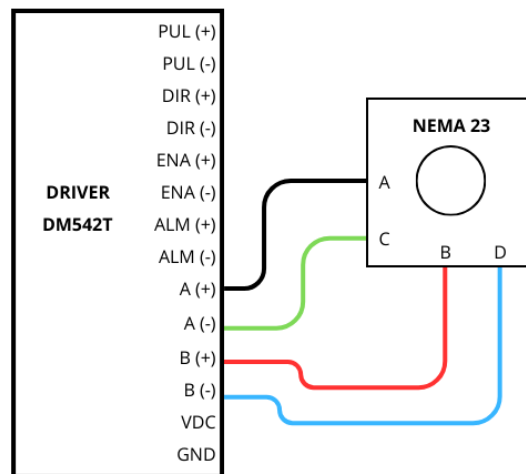


Figura 5.3: Diagrama das ligações entre o *driver* DM542T e o NEMA 23 [65]

em alguns pontos havia ligações um pouco "soltas". Por isso, foi pertinente retificar as ligações mencionadas.

Ligação dos motores NEMA 34 aos *drivers* DM860I

Tal como nos NEMA 23, os motores NEMA 34 também têm dois pares de bobinas correspondentes às duas fases do motor, onde A e C, o primeiro par, está simbolizado a vermelho e verde, e B e D, o segundo par, está designado a amarelo e azul, como se observa no diagrama da **Figura 5.4** [66].

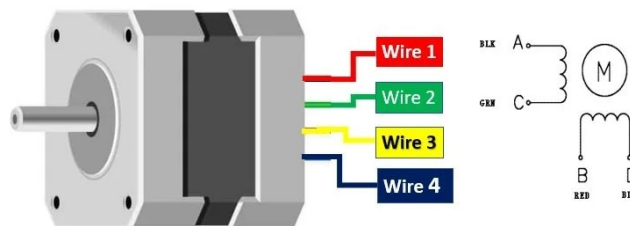


Figura 5.4: Diagrama das ligações possíveis ao motor NEMA 34 [66]

Do mesmo modo que acontece com o *driver* DM542T, o *driver* DM860I tem as mesmas ligações possíveis às fases do motor: A+, A-, B+ e B-, como se pode visualizar na **Figura 4.16a** do **Subcapítulo 4.1.3**. Assim, a maneira correta de fazer a ligação entre o *driver*, DM869I, e o motor, NEMA 34, é ligar A e C a A+ e A-, respetivamente, e ligar B e D a B+ e B-, respetivamente, como se observa no esquema da **Figura 5.5** [60].

Tal como referido para os NEMA 23, para o sistema funcionar corretamente é necessário estabelecer a ligação entre cada um dos motores NEMA 34 e cada um dos

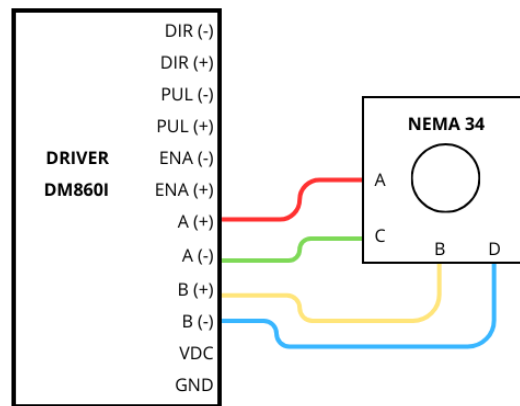


Figura 5.5: Diagrama das ligações entre o *driver* DM860I e o NEMA 34 [66]

respetivos *drivers*. Mais uma vez, foi necessário rever as ligações entre cada motor e *driver*, identificando alguns pontos de ligação frágeis. Por isso, foi essencial ajustar e corrigir as mesmas para garantir um funcionamento adequado deste conjunto.

Retificação das ligações entre os motores e os *drivers*

Como referido nos **Subcapítulos** anteriores, 3.5.1 e 3.5.1, havia ligações soltas entre os motores e os respetivos *drivers* que podiam causar falhas intermitentes no envio de sinais de controlo e gerar movimentos inconstantes dos motores, nomeadamente falhas na precisão e perda de passos.

Para corrigir este problema e garantir que o sistema funcionava de forma estável, foi realizado o seguinte:

- As ligações que se encontravam estabelecidas foram verificadas e reapertadas e, em alguns casos, foi utilizado isolamento adequado nas ligações para garantir que os fios ficassem bem fixados nos terminais dos *drivers* e motores.
- Foram realizados testes de continuidade elétrica, com um multímetro, para certificar que todas as ligações estavam estabelecidas corretamente e sem interrupções ou com uma resistência elevada, o que garantiu que a corrente e os sinais fossem transmitidos eficientemente entre os componentes.

Estes processos permitiram que os sinais de controlo dos *drivers* fossem recebidos corretamente pelos motores, levando a que a máquina operasse de modo estável durante a maquinagem da peça.

Ajustes aos *drivers*

Depois de serem verificadas todas as ligações entre os componentes referidos, motores e *drivers*, foi de extrema relevância ajustar os *drivers* para serem configurados corretamente para os motores em causa. Estas configurações consistiram em:

- Ajustar a corrente nos *drivers* para ir de encontro às especificações dos motores;
- Definir a resolução teórica do micropasso, que depende da precisão desejada. Através da seguinte **Equação 5.1**, é possível encontrar este valor que indica quantos passos o motor precisa realizar para o eixo em causa se mover 1 mm:

$$\frac{\text{passos}}{\text{mm}} = \frac{(\text{passos/revolução}) * \text{micropassos}}{\text{mm/revolução}} \quad (5.1)$$

Nos motores utilizados nesta máquina os passos por revolução são 200, com ângulo de 1,8°, tal como referido no **Subcapítulo 4.1.3**, logo, foi necessário encontrar os micropassos utilizados pelo fabricante e calcular os mm por revolução para cada passo:

- O eixo Z, utiliza um NEMA 23, que movimenta o eixo ao longo de duas guias com o auxílio de um fuso de esferas com rosca com passo 2 mm. Assim, através da **Tabela B.1** do **Apêndice B.6**, é possível perceber que existem disponíveis nestes *drivers* micropassos de 2 mm, logo a equação corresponde para o cálculo dos passos/mm para este eixo está representação pela **Equação 5.2** [67]:

$$\frac{\text{passos}}{\text{mm}}_{\text{eixoZ}} = \frac{200 * 2}{2} = 200\text{passos/mm} \quad (5.2)$$

- O eixo X, utiliza dois NEMA 34, que movimentam o eixo verticalmente, através de duas polias de igual dimensão, ligadas aos motores. Como as polias intersectam a correia é preciso contar os dentes da polia nesta intersecção, que corresponde a oito dentes. Como a correia tem um passo de 8 mm, e pela **Tabela B.2** do **Apêndice B.6**, percebe-se que existem disponíveis nestes *drivers* micropassos de 16 mm, então obtém-se a seguinte **Equação 5.3**:

$$\frac{\text{passos}}{\text{mm}}_{\text{eixoX}} = \frac{200 * 16}{8 * 8} = 50\text{passos/mm} \quad (5.3)$$

- O eixo A, utiliza um NEMA 23, que movimenta o eixo de rotação. Como existe falta de informação relativamente às polias e correias que constituem este sistema, o valor dos micropassos tem de ser ajustado. Sabe-se que a polia

do motor tem catorze dentes e a polia passiva trinta e quatro, com diâmetro de 110 mm, logo obtém-se a **Equação 5.4**:

$$\frac{\text{passos}}{\text{mm}} \text{eixoA} = \frac{200 * 8 * 34}{14 * \pi 110} = 11,24 \text{passos/mm} \quad (5.4)$$

Os valores apresentados nas **Equações 5.2, 5.3 e 5.4**, apesar de teóricos, tiveram de ser testados na máquina para perceber se o funcionamento e precisão da maquina-gem é adequado. Além disso, como estes parâmetros se encontravam *standardizados* na configuração do Grbl, foi necessário alterar os mesmos, inserindo os novos valores calculados nos comandos \$100, \$102 e \$103. A definição dos novos valores bem como os testes efetuados para perceber se os valores obtidos se encontram nos parâmetros adequados, encontra-se mostrado adiante no **Subcapítulo 5.2**.

5.1.2 Ligação entre as fontes de alimentação e os *drivers*

Após a verificação e correção das ligações entre os motores e respetivos *drivers*, foi, igualmente, necessário serem verificadas as ligações entre as fontes de alimentação S-350-60 e os respetivos *drivers*. Para tal foi essencial serem consultados os *datasheet* dos *drivers* e da fonte de alimentação e analisar a secção específica referente às ligações de corrente, para identificar as ligações adequadas entre estes componentes.

Como mostrado nas Figuras 4.16 e 4.18, no **Subcapítulo 4.1.3**, os *drivers* têm os terminais voltagem corrente contínua (VDC) e *Ground* (GND), enquanto a fonte de alimentação apresenta as saídas V+ e V-. Assim, devem ser realizadas as seguintes ligações [63] [60] [61]:

- A saída V+ da fonte de alimentação deve estar ligada ao terminal VDC dos *drivers*;
- A saída V- da fonte de alimentação deve estar ligada ao terminal GND dos *drivers*.

Na máquina em estudo, e como mencionado anteriormente na **Tabela 4.1** do **Subcapítulo 4.1**, são utilizadas três fontes de alimentação para fornecer energia a cinco *drivers*. Este facto deve-se às exigências de potência e corrente dos dois motores NEMA, controlados por estes *drivers*.

Quando comparados com os NEMA 23, os NEMA 34, normalmente têm um maior consumo de corrente e por isso precisam de mais energia. Como são fisicamente maiores têm um torque mais elevado, logo precisam de uma alimentação elétrica mais robusta para operar de forma eficiente.

Neste projeto, os NEMA 34 são utilizados para movimentar o eixo X, que controla o movimento do *spindle* e da estrutura à qual este se encontra acoplado. Como

esta estrutura tem um peso considerável, é necessária uma força significativa para movimentá-la, justificando o uso de motores com torque elevado. Além disso, como os dois motores NEMA 34 são usados em simultâneo para movimentar a estrutura de forma estável e sincronizada, faz com que seja ainda mais pertinente a utilização de fontes de alimentação separadas para cada um dos *drivers* destes motores [60] [57]

Por outro lado, os NEMA 23 como são menores e necessitam de menor corrente e potência para operar, podem ter partilhada a fonte de alimentação, pois a corrente total necessária para os três motores encontra-se nos limites de uma única fonte, e por isso pode ser usada sem a sobrecarregar. Neste caso, uma fonte de alimentação está ligada aos três *drivers* destes motores [61] [56].

Como cada uma das fontes de alimentação tem três saídas V+ e três V-, como se verifica na **Figura 4.18** do **Subcapítulo 4.1.3**, é possível realizar-se adequadamente as ligações entre os componentes fonte de alimentação e *drivers*. Assim, é possível serem observadas as ligações mencionadas entre os *drivers* e as fonte de alimentação, nas **Figuras 5.6** e **5.7**.

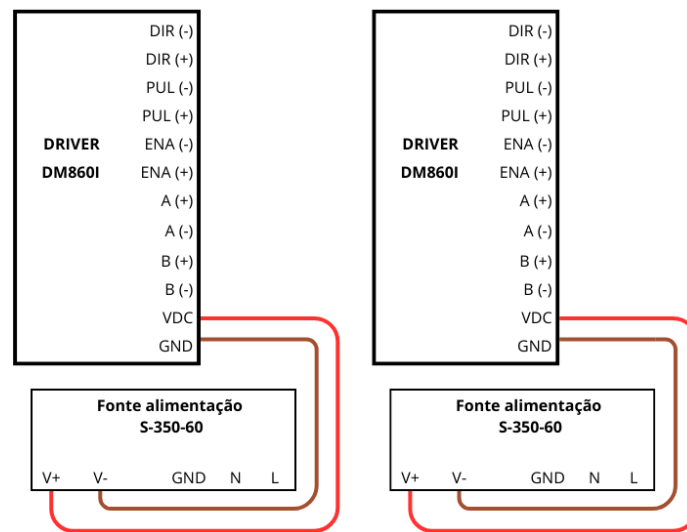


Figura 5.6: Ligações entre os *drivers* DM860I e duas fontes de alimentação S-350-60 [60] [63]

Assim como foi necessário ajustar as ligações entre os motores e os seus respetivos *drivers*, foi, do mesmo modo, essencial rever as ligações entre as fontes de alimentação e os *drivers*. Durante esta etapa, foram identificadas alguns pontos de ligação frágeis nas ligações às fontes de alimentação. Por isso, foi fundamental ajustar e corrigir as mesmas para garantir que a energia fosse fornecida de forma consistente e segura, assegurando o funcionamento adequado do sistema.

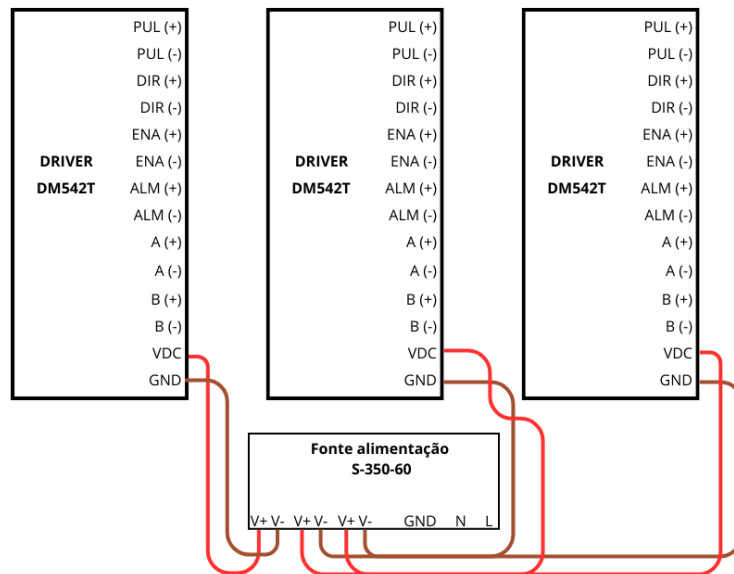


Figura 5.7: Ligações entre os *drivers* DM542T e uma fonte de alimentação S-350-60 [61] [63]

Retificação das ligações entre as fontes de alimentação e os *drivers*

Quando existem ligações mal estabelecidas nas fontes de alimentação podem ocorrer variados problemas como quedas de tensão, fornecimento irregular de corrente ou até mesmo falhas completas de energia, que podem resultar na perda de precisão dos motores ou danos nos componentes eletrônicos.

A retificação deste problema, garante que o sistema funciona de forma estável, assim, procedeu-se da seguinte forma:

- Foi feita uma revisão completa em todas as ligações entre os componentes referidos, dando ênfase aos terminais V+ e V-. Certificou-se de que todos os fios estavam devidamente fixados e conectados, através dos parafusos de fixação existentes nos componentes.
- Depois do ajuste das ligações, foram realizados testes de saída de tensão e fornecimento de corrente em cada *driver*, com o recurso de um multímetro, para assegurar que cada *driver* recebia a quantidade exata de energia necessária para o funcionamento dos motores.
- Para evitar problemas futuros, optou-se por utilizar ligações com bitola apropriada para suportar a corrente exigida pelos motores, para o sistema operar sem aquecimento excessivo ou sobrecarga.

Estes passos garantiram que a energia fosse fornecida de maneira consistente e segura, reduzindo o risco de falhas de energia.

Para perceber se os motores se encontravam a funcionar corretamente, foram realizados testes a cada um dos motores, isoladamente, recorrendo ao Arduino Mega 2560. Os resultados destes testes encontram-se enumerados no **Subcapítulo 5.2**.

5.1.3 Ligações entre o controlador e os *drivers*

Como referido no **Subcapítulo 4.1.3**, o Arduino Mega 2560 é o controlador utilizado neste sistema, sendo responsável por gerar os sinais de controlo para os *drivers* dos motores. Estes sinais são enviados através dos pinos de controlo do Arduino, visíveis na **Figura B.4** do **Apêndice B.3**, configurados para cada eixo.

No **Subcapítulo** anterior, 5.1.1, é referido que a cada motor NEMA está associado um *driver* específico, de acordo com as suas especificações de operação. As ligações disponíveis em ambos os *drivers* acontecem, essencialmente, através de quatro entradas principais:

- PUL+ e PUL-, responsáveis pelos pulsos de passo (*steps*), controlando a quantidade de passos que cada motor deve executar. Cada pulso recebido pelo *driver* representa um movimento incremental, cuja fração de um passo total depende da configuração dos micropassos realizada nos *drivers*, referida no **Subcapítulo 5.1.1**.
- DIR+ e DIR-, responsáveis pela direção do movimento do motor, determinando em que sentido, horário ou anti-horário, a rotação do motor acontece. As mudanças no estado lógico deste sinal invertem o sentido de rotação, o que permite o controlo completo do deslocamento dos eixos [60] [61].

A última fase de verificação de ligações prende-se com as ligações das entradas referidas acima, PUL+, PUL-, DIR+ e DIR-, aos pinos de controlo do Arduino, que envia os sinais de passo e direção para os motores.

Como é necessário ligar os vários *drivers* ao controlador, é indispensável utilizar-se uma *proto-board* para o efeito, que já se encontra instalada no sistema da máquina. A utilização deste componente facilita a ligação dos cinco *drivers* do sistema, bem como a organização dos sinais de controlo, permitindo que o Arduino consiga gerir eficientemente os motores. Além disso, a alimentação de 5 V e GND para os sinais PUL e DIR é crucial para garantir que os sinais sejam corretamente interpretados pelos *drivers*, assegurando a consistência no controlo de movimento dos eixos.

Adicionalmente, o interruptor de limitação de movimento (*limit switch*), desempenha um papel crítico na segurança da máquina, pois ao ser ligado a um pino digital do Arduino garante que o movimento dos eixos é interrompido caso o motor atinja o final do curso físico, prevenindo possíveis danos mecânicos à máquina.

Após a leitura do *datasheet* do Arduino e verificação das ligações possíveis de efetuar entre este controlador e os *drivers*, obteve-se a **Figura B.6**, presente no

Apêndice B.5, que mostra as ligações estabelecidas entre todos os componentes do sistema. Através do circuito desta **Figura**, conseguem-se perceber, resumidamente, as seguintes ligações:

- Os terminais PUL- e DIR- encontram-se ligados aos pinos digitais do Arduino;
- Os terminais PUL+ e DIR+ encontram-se ligados perto do pino 5V, na *proto-board*;
- *Limit switch*.

Estas ligações enumeradas encontram-se pormenorizadas nos **Subcapítulos** seguintes 5.1.3 e 5.1.3.

Ligação dos *drivers* do eixo A e Z ao controlador

Como referido previamente, no **Subcapítulo** 4.1.1, os eixos A e Z são movimentados através dos motores NEMA 23, controlados pelos *drivers* DM542T. Para haver comunicação entre estes *drivers* e o Arduino e como verificado na **Figura** B.6 do **Apêndice B.5**, são necessárias efetuar as seguintes ligações:

1. Ligações do eixo A (dois motores e dois *drivers*):
 - PUL+ e DIR+ ligados ao pino 5 V;
 - PUL- ligado ao pino D25;
 - DIR- ligado ao pino D34;
2. Ligações do eixo Z (um motor e um *driver*):
 - PUL+ e DIR+ ligados ao pino 5 V;
 - PUL- ligado ao pino D24;
 - DIR- ligado ao pino D35; [61] [59]

Como os sinais PUL+ e DIR+ necessitam de uma tensão positiva estão ligados ao pino de 5 V no Arduino, para fornecer a tensão necessária para os *drivers* funcionarem corretamente.

Por outro lado, os sinais de PUL- e DIR- são frequentemente ligados ao GND ou a pinos digitais, dependendo da configuração desejada. Neste caso, optou-se por ligar estes sinais a pinos digitais do Arduino, pois esta configuração permite que os sinais de pulso e direção sejam geridos digitalmente, oferecendo maior controlo sobre o comportamento dos motores.

Tal como realizado para os motores, foi importante perceber se as ligações entre o controlador e os *drivers* produzia movimento nos eixos referidos. Assim, foram

realizados testes em cada um dos eixos, A e Z, isoladamente, recorrendo ao Arduino Mega 2560. Os resultados destes testes encontram-se enumerados no **Subcapítulo** 5.2.

Ligação dos *drivers* do eixo X ao controlador

Por outro lado, e como mencionado previamente no **Subcapítulo** 4.1.1, os eixos X são movimentados através dos motores NEMA 34, controlados pelos *drivers* DM860I. Do mesmo modo que acontece com os *drivers* dos eixos A e Z, para acontecer a comunicação entre estes *drivers* e o Arduino e como verificado na **Figura** B.6 do **Apêndice** B.5 são necessárias efetuar as seguintes ligações dos dois motores e dois *drivers*:

- PUL+ e DIR+ ligados ao pino 5 V;
- PUL- ligado ao pino D22;
- DIR- ligado ao pino D37; [60] [59]

Como referido no **Subcapítulo** anterior, 5.1.3, as ligações dos sinais mencionados foram realizadas conforme os movimentos descritos anteriormente.

Tal como referido no **Subcapítulo** anterior, 5.1.3, foram realizados testes neste eixo recorrendo ao Arduino Mega 2560. Os resultados destes testes encontram-se enumerados no **Subcapítulo** 5.2.

5.1.4 Ligações entre o controlador, o *firmware* e o *software* de controlo

O processo de implementação do sistema de controlo baseia-se na ligação em série entre o controlador, previamente equipado com o *firmware* Grbl, e um canal de comunicação, responsável por enviar e receber dados do mesmo. Esta ligação realizada com o PC, habitualmente acontece através de um cabo USB, e permite a troca de informação entre estes dois sistemas.

Tal como referido nos **Subcapítulos** anteriores, para perceber se as ligações entre o *firmware* e o controlador se encontram bem definidas realizaram-se uma série de processos. Estes processos incluem a verificação do *firmware* definido no controlador e a sua versão, bem como o estabelecimento da ligação entre o controlador e o *software* de controlo da máquina. Estes passos são enumerados seguidamente.

Comunicação entre o *firmware* Grbl e o Arduino Mega 2560

Para verificar se o *firmware* Grbl estava corretamente definido no controlador Arduino Mega 2560, foi necessário seguir alguns passos utilizando um *Integrated Development Environment* (IDE), como o Arduino IDE. Este *software* é amplamente utilizado devido à sua compatibilidade direta com os MCU da gama Arduino, incluindo o Mega 2560, e pela facilidade de utilização, pois já tem instaladas bibliotecas pré-definidas e mapeamento de pinos adequado para facilitar o desenvolvimento de projetos com o Grbl. Além disso, o IDE permite carregar o *firmware* diretamente no controlador via USB e monitorizar a comunicação em tempo real através do *Serial Monitor*, simplificando o processo de teste e validação do sistema.

Deste modo, após a ligação da porta USB do PC ao controlador, utilizou-se o Arduino IDE para estabelecer corretamente a comunicação entre o Arduino e Grbl, configurando a comunicação serial no Grbl. Antes de ser verificado se a plataforma Grbl já estava instalada no controlador, definiu-se uma taxa de transmissão (*baud rate*) pré-definida de 115200 bps, uma vez que este é o padrão de velocidade para comunicação com o Grbl. A configuração escolhida permite a transmissão rápida o suficiente, especialmente, em aplicações CNC e na maioria dos PCs. Além da *baud rate*, a configuração do Grbl inclui oito bits de dados, sem paridade, e um bit de pausa ou *stop bit*, garantindo a transmissão de dados corretamente entre o Arduino e o PC. Esses parâmetros são comuns neste tipo de comunicações e confirmam que o PC e o Arduino leiam os dados transmitidos corretamente. Habitualmente este valor de *baud rate* já se encontra configurado no Grbl, caso contrário é necessário configurá-lo diretamente no código-fonte do *firmware* [68].

O PC envia os comandos G-code para o Arduino através do *Serial Monitor*, no entanto, esta comunicação com o *firmware* só é possível se a linguagem utilizada for a mesma que o controlador requer. Assim, muitas vezes é utilizada a linguagem ASCII para o efeito, onde o Grbl a interpreta para controlar os motores e a máquina CNC. Quando o operador envia um comando válido para o Grbl este processa-o e retorna uma resposta através do canal de comunicação, confirmando a ação ou enviando dados de retorno. A linguagem ASCII é usada para essa comunicação, o que facilita a troca de dados entre o controlador e o *firmware*. As funções ou listas de funções pré-definidas para cada comando, são as mesmas observadas na **Tabela C.4** do **Subcapítulo C.2**.

Quando a ligação em série se encontra estabelecida corretamente, o Grbl envia uma mensagem de início pelo canal de comunicação, como se pode verificar pela **Figura 5.8** [69], que indica a versão do Grbl instalada, bem como o comando de ajuda \$.

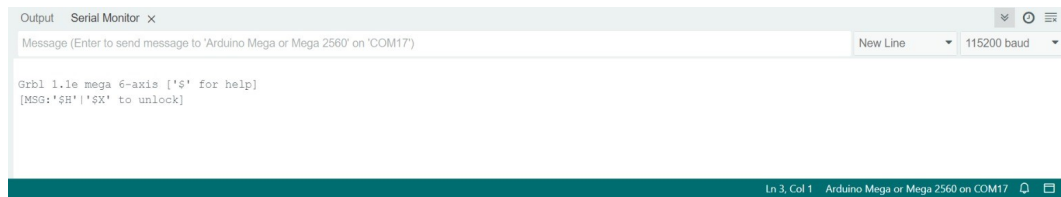


Figura 5.8: Mensagem de início enviada pelo Grbl, visível no *Serial Monitor* do Arduino IDE

Além da mensagem indica pela **Figura 5.8**, aquando da ligação estabelecida, também é possível ver no *Serial Monitor* as configurações do Grbl, utilizando o comando \$\$, como mostra a seguinte **Figura 5.9**.

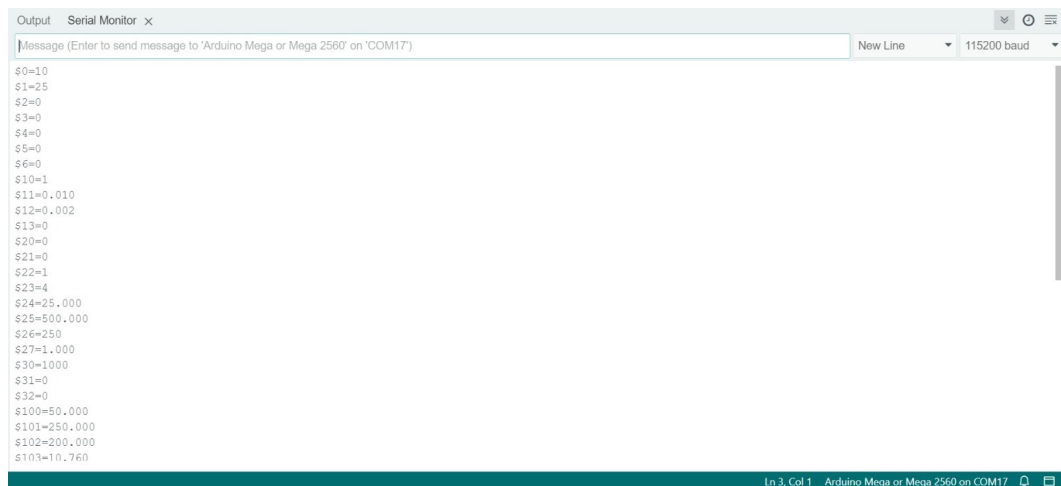


Figura 5.9: Parâmetros do Grbl configurados no Arduino, visíveis no *Serial Monitor* do Arduino IDE

O *firmware* Grbl foi originalmente projetado para o Arduino Uno que tem menos pinos, contudo, o Mega 2560 por ter mais pinos pode ser utilizado para expandir as funcionalidades, como adicionar mais eixos de atuação, como é o caso. A correspondência dos pinos entre o Grbl e o Arduino Mega 2560 pode ser feita conforme os motores necessários controlar e outras funcionalidades da CNC.

O Grbl-Mega-5X, como referido no **Subcapítulo 4.1.3**, é uma versão modificada do Grbl desenvolvida para quando é necessário controlar até cinco eixos de movimento em simultâneo. Logo, é necessário instalar esta versão no controlador e configurar os pinos no ficheiro adequado, podendo ser consultado o código utilizado no mesmo no **Apêndice D.1** [70].

Comunicação entre o *firmware* Grbl e o Grbl Controller

Para facilitar o envio e receção de dados entre o operador e o sistema CNC, normalmente utilizam-se *softwares* de controlo CNC com uma interface gráfica que

funciona como um intermediário entre o operador e o *firmware* Grbl. Existem várias opções de *softwares open source* disponíveis no mercado, compatíveis com diversos SO, como Windows, MacOS ou Linux. Um exemplo bastante utilizado é o Grbl Controller, já utilizado com esta máquina, que oferece uma interface gráfica simplificada, permitindo ao operador visualizar e controlar as operações da máquina CNC com facilidade.

Como mencionado anteriormente no **Subcapítulo 4.1.3**, o Grbl Controller permite ao operador visualizar, editar e enviar comandos G-code diretamente para o *firmware* Grbl. A interface gráfica torna o processo mais intuitivo, uma vez que é possível ver o caminho de corte que será seguido pelos motores e fazer ajustes em tempo real. O *software* também oferece informações detalhadas sobre o estado atual da máquina, como a posição dos eixos, a velocidade do *spindle* (se automatizado), e o progresso da operação.

Além disso, permite ao operador monitorizar o retorno de dados do Grbl, como mensagens de erro, alertas ou confirmações de execução dos comandos o que ajuda a garantir que os comandos são processados corretamente e que a operação está a ser executada conforme o planeado. O operador também pode ajustar parâmetros diretamente no *software*, como velocidades de deslocamento, taxas de alimentação e outros parâmetros críticos.

Deste modo, e para facilitar a interação entre o *firmware* e o controlador, utilizou-se este *software*. Quando se abre o *software* pela primeira vez o ecrã visível é o que aparece na **Figura 4.20**, do **Subcapítulo 4.1.3**. Depois disto, e como realizado no Arduino IDE, é necessário ser configurada a *baud rate* para 115200 para ser possível a comunicação entre o controlador e o *firmware* acontecer. Posto isto, aparece uma mensagem com a versão do Grbl instalada, bem como o comando de ajuda \$, como se observa na **Figura 5.10**.

Quando o Grbl recebe um comando G-code compatível, interpreta-o e realiza a ação correspondente. Estas ações podem variar desde movimentos dos eixos, como G1 X10 Z20 F300 para mover os eixos X e Z, ou até mesmo comandos mais avançados, como a definição de *home* ou a configuração de parâmetros da máquina. O *firmware* também é responsável por manter a comunicação com o PC, enviando mensagens de estado ou erros para a interface gráfica.

Se por acaso ocorrer uma falha de energia no PC, como falta de bateria, enquanto esteja a decorrer uma operação na máquina, a mesma continua a funcionar a não ser que seja manualmente cortada a energia. Contudo, não é possível retornar a ligação ao canal previamente estabelecido e o operador deixa de ter controlo sobre a operação que está a ser realizada. Esta situação deve-se ao facto de os sistemas serem separados uma vez que o *firmware* não é ativado até haver energia na máquina.

Uma das limitações encontradas durante a utilização deste sistema é a falta de

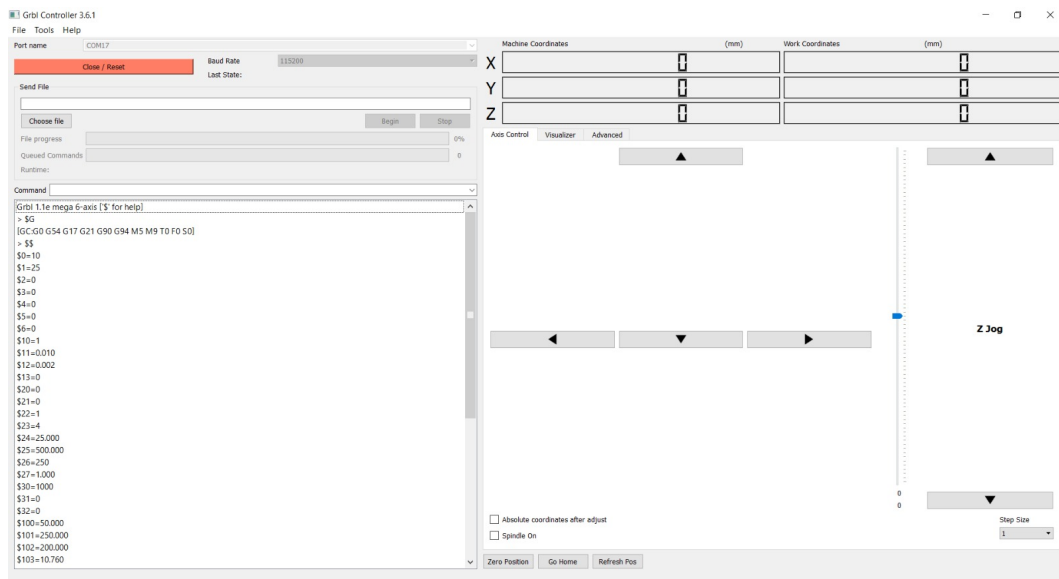


Figura 5.10: Mensagem de início enviada pelo Grbl, através da interface gráfica do Grbl Controller

recuperação automática em caso de falhas de energia ou falhas repentinas do PC. No caso de uma falha de energia no PC, como bateria fraca ou corte de eletricidade, a máquina CNC continua a operar até a energia ser cortada manualmente. No entanto, o canal de comunicação entre o Grbl e o PC é perdido, e não é possível restabelecer a ligação com o canal previamente ativo. Isto significa que o operador perde a capacidade de monitorizar a operação em curso e enviar novos comandos, o que faz com que ocorra uma situação de "falha no controlo" sobre a máquina.

Esta limitação decorre do fato de o Grbl operar de maneira independente do PC após o envio dos comandos iniciais. O *firmware* Grbl continua a executar o G-code até o programa ser concluído ou até o sistema ser desligado manualmente, pois a máquina e o PC são sistemas independentes.

5.2 Testes de validação e ajustes finais

Depois de verificadas todas as ligações acima referidas no **Subcapítulo 5.1**, foram realizados alguns testes no sentido de perceber se o sistema CNC se encontrava a funcionar corretamente.

Em primeiro lugar, foi importante perceber se os motores se encontram a funcionar corretamente. Em seguida, foi necessário entender se o *firmware* se encontrava configurado no Arduino e se era possível a ligação entre ambos. Por fim, e após estabelecida a ligação entre o controlador e *firmware*, foi verificado se era possível estabelecer-se a ligação do *firmware* com o *software* Grbl Controller para ser mais

fácil visualizar o sistema em tempo real e existir uma interface gráfica para controlo do sistema CNC.

5.2.1 Testes aos movimentos dos motores

Como referido no **Subcapítulo 5.1.2**, para perceber se os motores se encontravam a funcionar corretamente, foi necessário realizarem-se testes a cada um dos motores isoladamente.

Para a realização dos testes referidos foi necessário serem retiradas as ligações existentes entre o Arduino e os *drivers* para melhor perceção e configuração entre estes dois componentes. Assim, foram realizados dois tipos de configurações entre o Arduino e os *drivers*, uma para os *drivers* referentes aos motores NEMA 23 e outra para os motores NEMA 34.

Configuração dos pinos para testar os motores NEMA 23

A configuração dos pinos para testar os motores NEMA 23 prendeu-se com a seguinte:

- Ligar o pino DIR+, PUL+ e ENA+ ao pino 5 V;
- Ligar o pino PUL- ao pino digital 24;
- Ligar o pino DIR- ao pino digital 35;
- Ligar o pino ENA- ao pino digital 28;

Para se poder testar cada um dos três motores NEMA 23 foi necessário substituir-se as ligações de cada um dos *drivers* no Arduino (PUL+, PUL-, DIR+ e DIR-). Além disso, foi mandatário proceder-se à ligação dos pinos ENA+ e ENA- dos *drivers* ao controlador, uma vez que estas ligações não se encontravam definidas, passo fundamental para habilitar os *drivers* a realizar os movimentos do motor.

Para testar a configuração realizada acima, foi necessário proceder-se à reconfiguração do controlador, recorrendo ao Arduino IDE através do código que se apresenta no **Apêndice D.1.1**.

Aquando do envio do código para o Arduino verificou-se que o motor realizou três rotações completas no sentido anti-horário e três rotações completas no sentido horário, com uma pausa de 5 s entre a inversão do sentido do movimento.

Configuração dos pinos para testar os motores NEMA 34

Os mesmos testes realizados aos motores NEMA 23 foram realizados no NEMA 34, diferindo na configuração dos pinos que se prendeu com a seguinte:

- Ligar o pino DIR+, PUL+ e ENA+ ao pino 5 V;
- Ligar o pino PUL- ao pino digital 32;
- Ligar o pino DIR- ao pino digital 37;
- Ligar o pino ENA- ao pino digital 26;

Tal como referido no **Subcapítulo** 5.2.1, para se poder testar cada um dos dois motores NEMA 34 foi necessário substituir-se as ligações de cada um dos *drivers* no Arduino (PUL+, PUL-, DIR+ e DIR-), bem como proceder-se à ligação dos pinos ENA+ e ENA- dos *drivers* ao controlador.

Para testar a configuração realizada acima, foi necessário proceder-se à reconfiguração do controlador, recorrendo ao Arduino IDE através do código que se apresenta no **Apêndice** D.1.2.

Aquando do envio do código para o Arduino verificou-se que o motor realizou três rotações completas no sentido anti-horário e três rotações completas no sentido horário, com uma pausa de 5 s entre a inversão do sentido do movimento.

5.2.2 Testes aos movimentos dos eixos

Como indicado nos **Subcapítulos** 5.1.3 e 5.1.3 para compreender se os eixos se encontravam a funcionar adequadamente, e após estabelecida a ligação entre o *firmware* e o *software* de controlo, um dos testes realizados foi enviar comandos pelo canal de comunicação do *software* para movimentar os eixos.

Assim, foram enviados os seguintes comandos, que podem ser observados na **Figura** 5.11:

```
1 G91
2 G0 X10
```

Com o envio deste comando, o Grbl deverá mover o eixo X 10 mm rapidamente, para a posição referida, e depois é enviada a confirmação da realização desse movimento para o Grbl Controller. Após ter sido inserido o comando referido, reparou-se que ocorreu movimento no eixo X de 10 mm, contudo, quando houve a tentativa de realizar o movimento inverso neste eixo, com o código G0 X-10 verificou-se que o movimento acontecia no mesmo sentido do anterior, indicando uma má configuração dos eixos ou dos *drivers* dos motores.

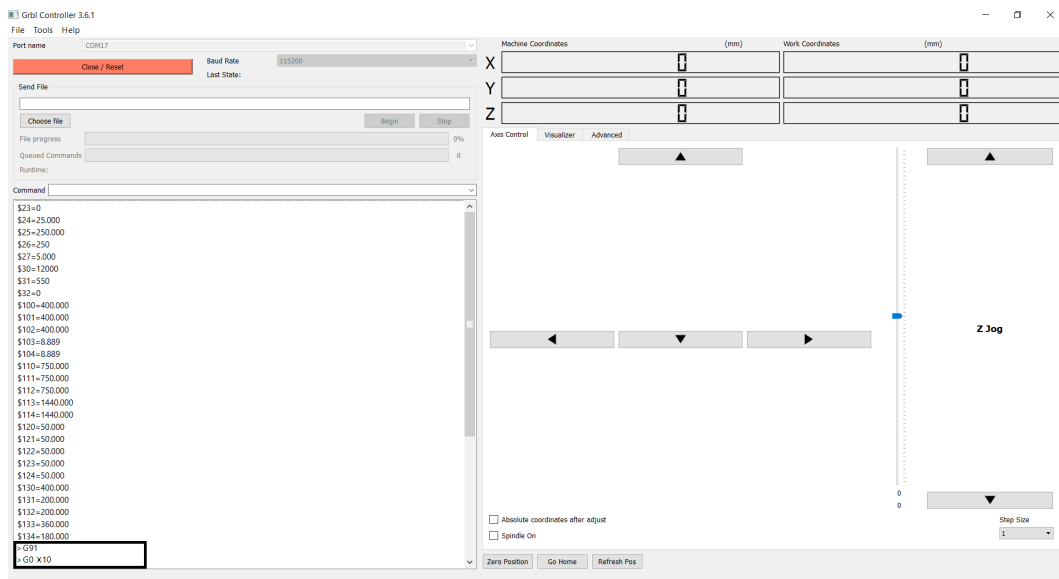


Figura 5.11: Comandos enviados para testar o movimento do eixo X

Reconfiguração dos pinos dos eixos da máquina CNC "QRC"

Como foi encontrado o problema acima mencionado e para verificar se os restantes eixos se encontravam a funcionar corretamente, realizaram-se outros testes semelhantes, para cada eixo. Após a realização de testes nos outros eixos, A e Z, verificou-se que no eixo A apenas um dos motores se encontrava em funcionamento, mais concretamente o da parte superior do eixo, e que no eixo Z, não ocorria qualquer tipo de movimento.

Assim, e para tentar solucionar o problema encontrado, procedeu-se ao remapeamento dos pinos entre o Grbl e o Arduino, através do código encontrado no **Apêndice D.1**.

Depois de ter sido realizada uma nova configuração dos eixos do sistema, e como a versão do Grbl instalada era mais recente do que a que se encontrava no Arduino, observou-se a mensagem no *Serial Monitor* do Arduino IDE, como se vê na **Figura 5.12**

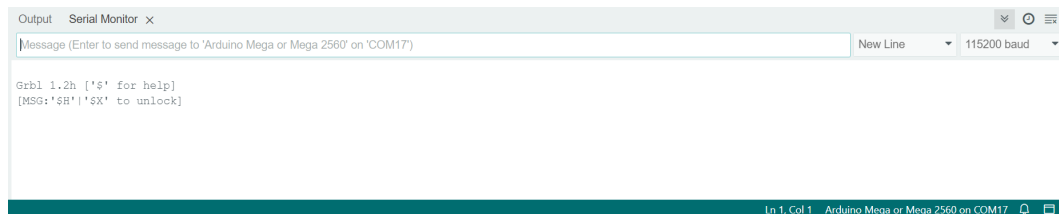


Figura 5.12: Nova mensagem de início enviada pelo Grbl, visível no *Serial Monitor* do Arduino IDE

Além da mensagem acima recebida no canal de comunicação do Arduino IDE, foi igualmente possível verificar-se as novas configurações do Grbl no Arduino, como se observa na **Figura 5.13**



```

Output Serial Monitor x
Message (Enter to send message to 'Arduino Mega or Mega 2560' on 'COM17') New Line 115200 baud

Grbl 1.2h ['$' for help]
[MSG:'$H'! '$X' to unlock]
[HLP:$$ $$ $D $G $I $N $X=val $Nx=line $J=line $SLP $C $X $H - ! ? ctrl-x]
ok
$0=10
$1=254
$2=0
$3=0
$4=0
$5=0
$6=0
$10=1
$11=0.020
$12=0.002
$13=0
$20=1
$21=0
$22=1
$23=0
$24=25.000
$25=250.000
$26=250
$27=5.000
$30=12000
$31=550
  
```

Figura 5.13: Novos parâmetros do Grbl configurados no Arduino, visíveis no *Serial Monitor* do Arduino IDE

Configuração dos limites máximos dos eixos

Além dos testes realizados aos movimentos dos eixos, foi necessário definir os limites de deslocamento máximo para os eixos X, Z e A, referentes ao tamanho da área de trabalho, de modo a prevenir que seja ultrapassado os limites físicos dos mesmos.

Uma vez que os valores definidos no Grbl se encontravam *standard*, como se pode verificar na **Figura 5.14**.

Deste modo, no canal de comunicação do Grbl Controller, foram inseridos os seguintes comandos próprios do Grbl, para o eixo X e Z, respetivamente, onde os valores inseridos dizem respeito à distância máxima que os eixos conseguem percorrer:

-
- 1 $\$130=700$
 - 2 $\$132=200$
-

Não foi necessário alterar o valor para o eixo Z uma vez que já se encontrava dentro do expectável.

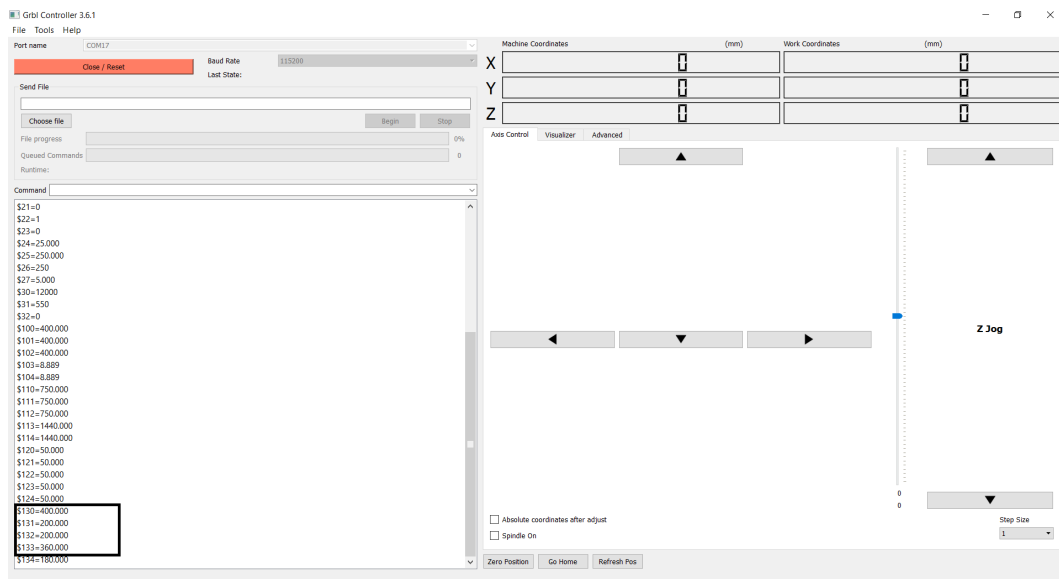


Figura 5.14: Comandos \$130, \$132 e \$133, referentes aos limites dos eixos

5.2.3 Testes realizados aos motores

As configurações presentes no Grbl são consideradas *default* pelo que é pertinente serem ajustados alguns valores de parametrização dos motores como os micropassos, velocidade e aceleração dos mesmos.

Assim, foram realizados testes neste sentido com base nos cálculos dos micropassos realizados anteriormente no **Subcapítulo 5.1.1**, e com base nas *datasheet* dos *drivers* e motores.

Testes aos micropassos dos motores

Um dos testes realizados aos motores foi configuração dos micropassos para cada um dos mesmos, uma vez que a configuração inserida era *standard*, como se pode analisar na **Figura 5.15**.

Deste modo, no canal de comunicação do Grbl Controller, foram inseridos os seguintes comandos próprios do Grbl, para o eixo X, Z e A, respetivamente, onde os valores obtidos para cada um dos eixos foram obtidos pelas **Equações 5.2, 5.3 e 5.4**, do **Subcapítulo 5.1.1**:

-
- 1 $\$100=50$
 - 2 $\$102=200$
 - 3 $\$103=11.24$
-

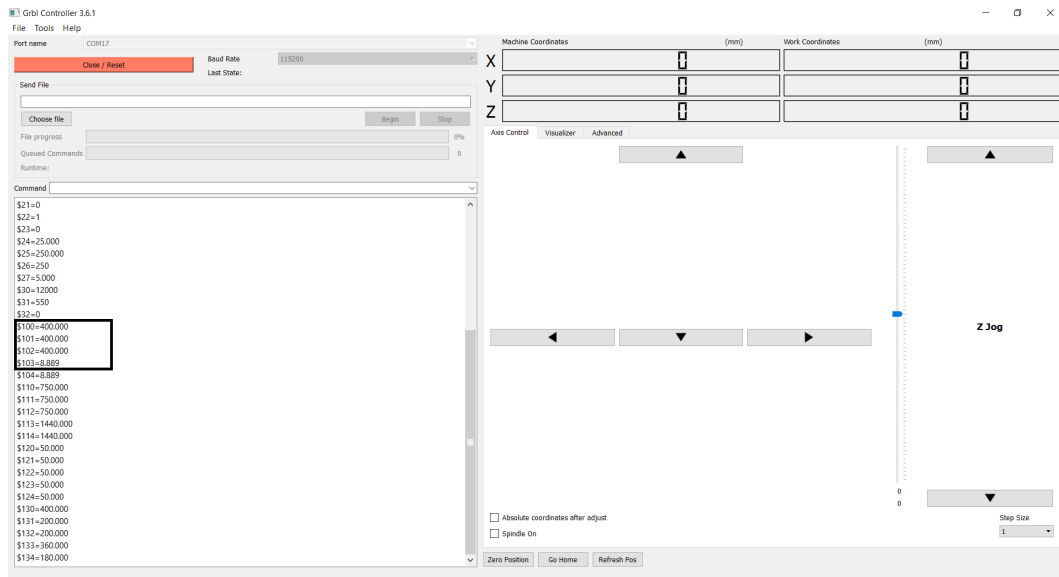


Figura 5.15: Comandos \$100, \$102 e \$103, referentes aos micropassos dos motores

Os ajustes efetuados apesar de teóricos representam uma calibração inicial, visando aproximar a resposta mecânica da máquina aos comandos digitais inseridos. Para se perceber se estes parâmetros se encontravam adequados teria sido importante terem sido realizados testes de precisão e repetibilidade que face ao tempo escasso para a realização do projeto não foi possível realizar.

Testes à velocidade dos motores

Outro dos testes necessários de realizar aos motores é a parametrização dos valores de velocidade máxima dos motores, uma vez que os mesmos também se encontram *standard* como se pode analisar na **Figura 5.16**.

Deste modo, no canal de comunicação do Grbl Controller, foram inseridos os seguintes comandos próprios do Grbl, para a velocidade dos motores dos eixos X, Z e A, respetivamente:

-
- 1 \$110=700
 - 2 \$112=700
 - 3 \$113=700
-

Os ajustes efetuados, como referido no início do **Subcapítulo**, tiveram como base os *datasheets* dos motores. Foram definidos os valores de 700 mm/min, valor intermédio das velocidades de funcionamento dos motores (500 a 1500 mm/min) uma vez o sistema CNC não apresenta demasiada robustez e por isso as velocidades

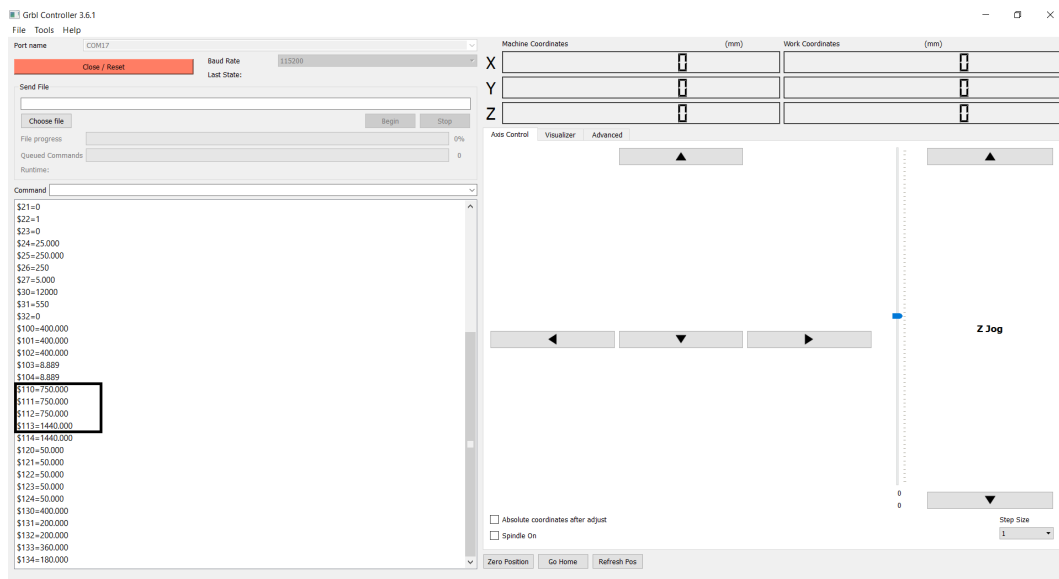


Figura 5.16: Comandos \$110, \$112 e \$113, referentes à velocidade dos motores

elevadas podem provocar vibrações excessivas na mesma. Apesar de se terem inseridos estes parâmetros e para se perceber se os mesmos se encontravam adequados, tal como nos parâmetros de micropassos, teria sido importante terem sido realizados testes de precisão e repetibilidade que face ao tempo escasso para a realização do projeto não foi possível realizar.

Testes à aceleração dos motores

Por fim, foi igualmente importante, realizar a parametrização dos valores de aceleração máxima dos motores, pois mais uma vez os mesmos também se encontravam *standard* como se pode observar na **Figura 5.17**.

Deste modo, no canal de comunicação do Grbl Controller, foram inseridos os seguintes comandos próprios do Grbl, para a aceleração dos motores dos eixos X, Z e A, respetivamente:

-
- 1 \$120=50
 - 2 \$122=50
 - 3 \$123=50
-

Os ajustes efetuados, como referido no início do **Subcapítulo**, tiveram como base os *datasheets* dos motores. Foram definidos os valores de 50 mm/s², valor mínimo da aceleração que o motor permite (50 a 200 mm/s²), pelas mesmas razões referidas acima, no **Subcapítulo 5.2.3** para a velocidade definida. Apesar de se

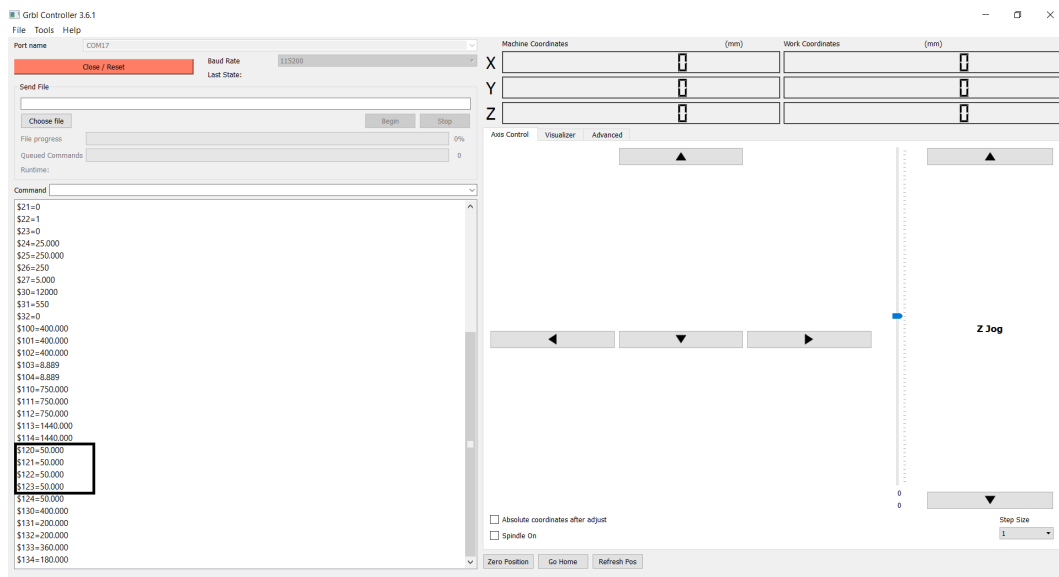


Figura 5.17: Comandos \$120, \$122 e \$123, referentes à aceleração dos motores

terem inseridos estes parâmetros e para se perceber se os mesmos se encontravam adequados, tal como nos parâmetros de micropassos e da velocidade de movimento dos motores, teria sido importante terem sido realizados testes de precisão e repetibilidade que face ao tempo escasso para a realização do projeto não foi possível realizar.

5.3 Fabrico de moldes na máquina CNC "QRC"

Para obter um molde positivo com a máquina CNC "QRC", é necessário seguir uma série de etapas, já apresentadas de forma geral no **Subcapítulo 2.2.2**. Desde a digitalização do membro até ao G-code ser gerado e a maquinação acontecer, cada fase do processo é crucial para garantir a precisão e a qualidade dos moldes. O fabrico de moldes positivos para dispositivos médicos exige uma abordagem detalhada e meticulosa, descrita, a seguir, com o exemplo de um molde positivo para uma prótese de membro inferior.

5.3.1 Digitalizar e retificar o molde positivo 3D

A digitalização do membro, para obtenção do molde positivo, é efetuada através de um *scanner* 3D, que neste caso tanto pode ser o Structure Sensor Mark I, acoplado a um IPAD, ou o Einscan Pro+. Ambos com tecnologias diferentes, mas com o objetivo de obter a forma do segmento do membro inferior desejado, como pode ser visualizado na **Figura 5.18**.



Figura 5.18: Digitalização de um membro residual com recurso a um *scanner* 3D

A obtenção deste segmento é realizada através de *software* próprio, do fabricante do *scanner* utilizado, que, após conclusão da digitalização, gera um ficheiro, normalmente guardado no formato .stl ou .obj.

Após ser conseguido o ficheiro, como referido no parágrafo acima, é necessário fazer o tratamento do mesmo, nomeadamente, retificações na malha 3D, para retirar artefatos provenientes da digitalização, bem como realizar algum tipo de retificação necessária para ser obtido o molde positivo 3D desejado. Para tal, é necessário ser utilizado um *software* CAD que permita modificar o ficheiro obtido. No caso em estudo, a empresa utiliza o Meshmixer, o SolidWorks ou o OrtenShape, sendo este último mais especificado para a área da Ortoprotesia, como é possível verificar-se na **Figura 5.19**.

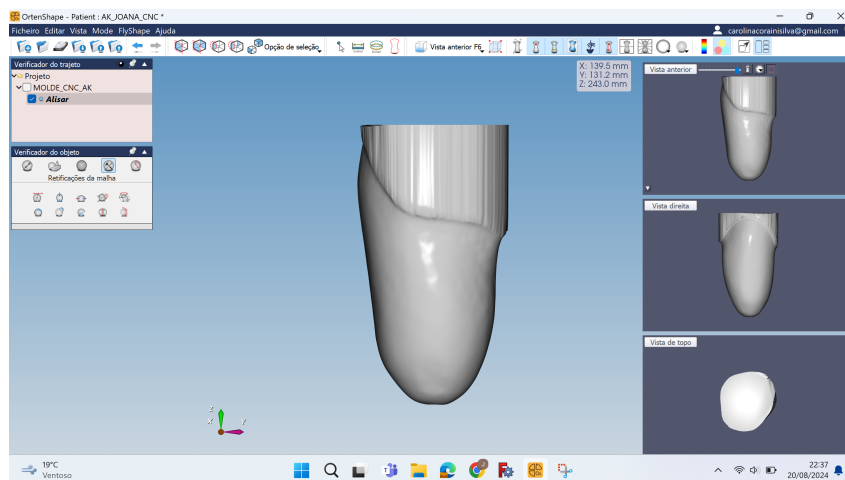


Figura 5.19: *software* CAD, OrtenShape para retificação do molde positivo

5.3.2 Orientar o molde positivo com as coordenadas da máquina "QRC"

Quando o molde 3D se encontra com a forma desejada, e antes do mesmo ser exportado para ser maquinado na máquina "QRC", é importante alinhá-lo segundo as coordenadas de movimento da mesma. Este aspeto pode ser conseguido no *software* utilizado para a retificação do molde positivo ou em outro *software* CAD.

Como mencionado no **Subcapítulo** 4.1.1, a máquina permite a maquinagem no eixo X, na vertical, no eixo Z, na horizontal, e no eixo A, através de rotação ou revolução. Como é comum que este tipo de moldes ser obtido por revolução, o bloco de PU deve ser fixado ao eixo A, possibilitando a maquinagem desta forma. Desta maneira, o comprimento do molde é definido pelo eixo X, enquanto os diâmetros são determinados pelo eixo Z, que corresponde à ferramenta de corte. Além disso, preferencialmente, e para minimizar vibrações e movimento indesejado da ferramenta de corte, o molde positivo deve ser orientado de modo que a sua extremidade de maior diâmetro fique paralela ao eixo X e perpendicular ao eixo Z.

Os aspetos referentes ao alinhamento dos eixos de coordenadas da máquina, são possíveis de observar na **Figura** 5.20, onde a extremidade de maior diâmetro do molde, está identificada com uma esfera azul (base). Comparando a **Figura** 5.19 e a **Figura** 5.20, na primeira os eixos de coordenadas não se encontram alinhado de modo que a maquinagem da peça seja realizada verticalmente, com o eixo Z paralelo à base do molde. Assim, foi necessário alterar a orientação do molde, verificando-se, na segunda imagem, que o eixo Z se encontra na posição correta, paralelo à base do molde.

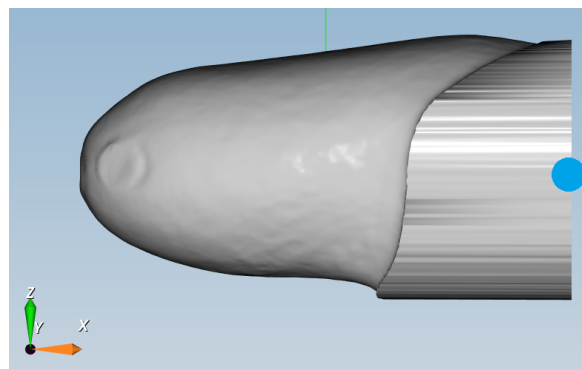


Figura 5.20: Alinhamento do molde do membro digitalizado, conforme os eixos de coordenadas da máquina

Outro aspeto importante para a correta maquinagem do molde, e como se pode ver na **Figura** 5.21, é o alinhamento do mesmo com o plano de corte, ou seja, com a extremidade da peça de fixação do bloco de PU, que corresponde a X0 ou à base do bloco.

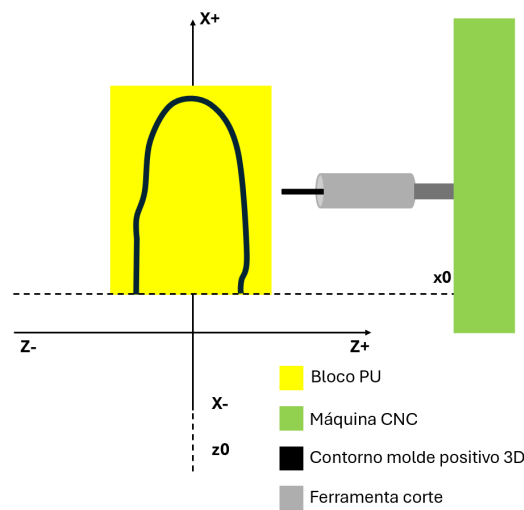


Figura 5.21: Alinhamento dos eixos para a maquinação, respeitando as características da máquina

Após serem realizados todos os processos de alinhamento quer do molde, quer do bloco de PU a ser maquinação, é necessário exportar o ficheiro no formato `.stl`, para o mesmo ser compatível no *software* CAM, que gera o G-code.

5.3.3 Gerar G-code

Para que qualquer máquina CNC possa maquinação uma peça, é necessário inserir um ficheiro com G-code, que pode ser obtido através de um *software* CAM, responsável por gerar o percurso que a ferramenta de corte, em conjunto com o movimento dos eixos, deverá realizar.

No caso da máquina em estudo, é utilizado o *software* DeskProto, por ser de fácil utilização, ter uma interface *userfriendly* e intuitiva, permitindo gerar o G-code facilmente, o que torna indicado mesmo para operadores que não tenham um conhecimento abrangente de como se pode gerar este tipo de ficheiros. Permite "criar" as máquinas, ferramentas de corte e controladores, conforme as configurações pretendidas pelo utilizador para estes aspetos. Com este *software* é possível serem obtidas configurações de máquinas CNC de até cinco eixos, incluindo os eixos de rotação, permitindo uma maior gama de aplicações.

Ainda, este *software*, tem disponível, assistentes de configuração que auxiliam o utilizador nas várias etapas de processamento da maquinação, reduzindo o risco de erros, tornando os processos mais rápidos. Além disso, existem versões deste *software* de valor acessível (até existe uma versão gratuita, com limitações de eixos de movimento), quando comparados com outros *softwares* de CAM para o mesmo efeito, e permite a utilização de uma variedade de formatos de ficheiros 3D como `.stl`.

Como o objetivo do *software* mencionado é gerar o G-code, para isto ser possível são importantes serem definidos alguns parâmetros e seguir alguns passos até serem obtidos os comandos para maquina o molde positivo na máquina CNC. É de referir, que para no estudo da máquina foi utilizada a versão 7.1 do *software*, apesar de ser uma versão *trial*, pois a versão utilizada, anteriormente, na empresa, já se encontra obsoleta e não permite inibir o eixo Y e habilitar o eixo A, de rotação.

Definição dos parâmetros da máquina e do pós-processador

Uma vez que a máquina utilizada se trata de uma máquina desenhada e concebida para ir de encontro às necessidades da empresa, a mesma não se encontra na biblioteca de máquinas do *software*. Assim, é necessário adicionar a máquina "QRC" introduzindo as características da mesma, nomeadamente, os valores da área de trabalho, em X e Z, a habilitação do eixo A e, ainda, a indicação do pós-processador utilizado, como se pode ver na **Figura 5.22**.

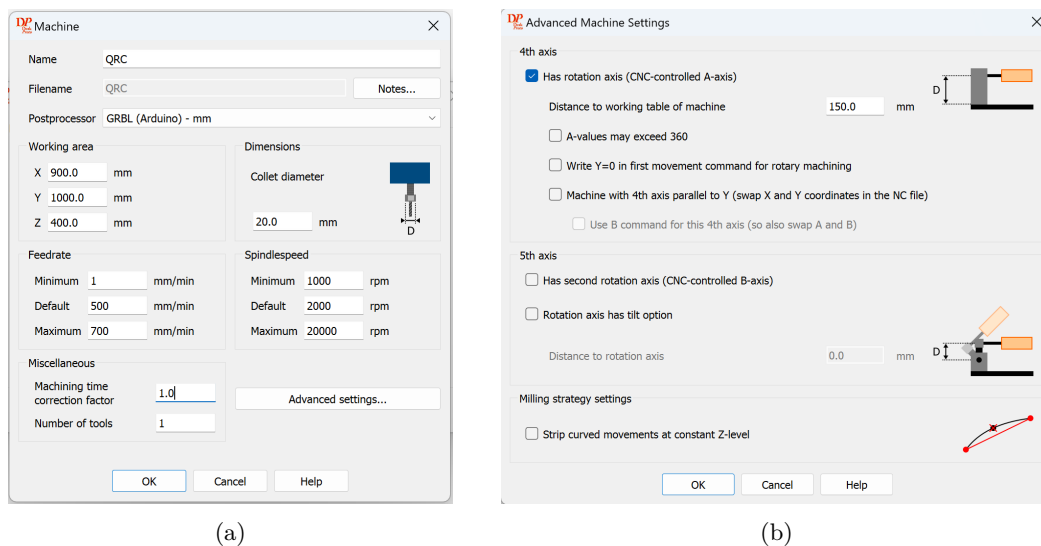


Figura 5.22: Configuração da máquina "QRC" (a) e habilitação do eixo A (b), no *software* DeskProto

Definir parâmetros da ferramenta de corte

Tal como na parametrização da máquina, é necessário adicionar a ferramenta de corte utilizada na máquina, pois como não existem dados do fabricante da mesma, não é possível encontrá-la na biblioteca. A **Figura 5.23** mostra a introdução dos valores referentes às características da ferramenta.

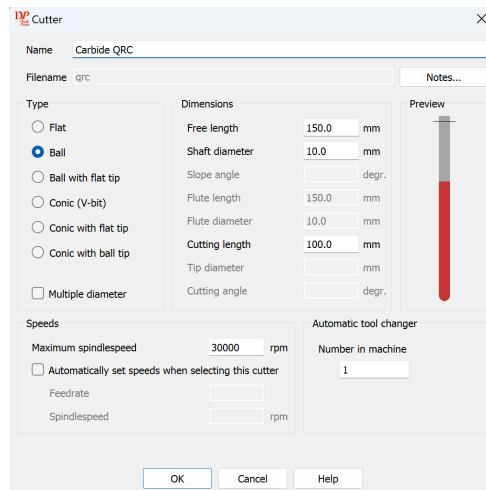


Figura 5.23: Configuração da ferramenta de corte, no *software* Desk-Proto

Configurar parâmetros de maquinagem e operação da máquina

Concluída a parametrização da máquina e da ferramenta de corte, os passos seguintes dizem respeito à configuração dos parâmetros de maquinagem e operação da máquina.

Em primeiro lugar, é fundamental importar o ficheiro *.stl* com a geometria da peça que se planeia maquinar. Após o ficheiro ser importado, é possível serem editadas algumas definições gerais, como o nome do projeto, a definição da máquina a utilizar, as operações que se pretendem realizar e, ainda, de extrema importância, por se tratar de uma máquina com eixo A, de rotação, habilitar a utilização deste eixo (inibindo o eixo Y).

Ainda, é necessário definir-se a dimensão do bloco a maquinar, para a ferramenta de corte saber a posição inicial e final do *path*. Normalmente, configura-se o bloco com as medidas de comprimento da peça e com metade das medidas de largura ou diâmetro do molde, sendo o valor máximo em Z, do molde positivo que se quer obter. Definem-se, assim, estas medidas para que a ferramenta não ultrapasse o zero do eixo Z. Também é definido o bloco como sendo customizado, com a forma de um paralelepípedo, com as dimensões de $260*150*150 \text{ mm}^3$, dando margem entre a peça e o mesmo, para permitir a fresagem sem haver falta de material para tal, sendo que o eixo A realiza um *path* de 360° .

As configurações mencionadas acima podem ser observadas na **Figura 5.24**.

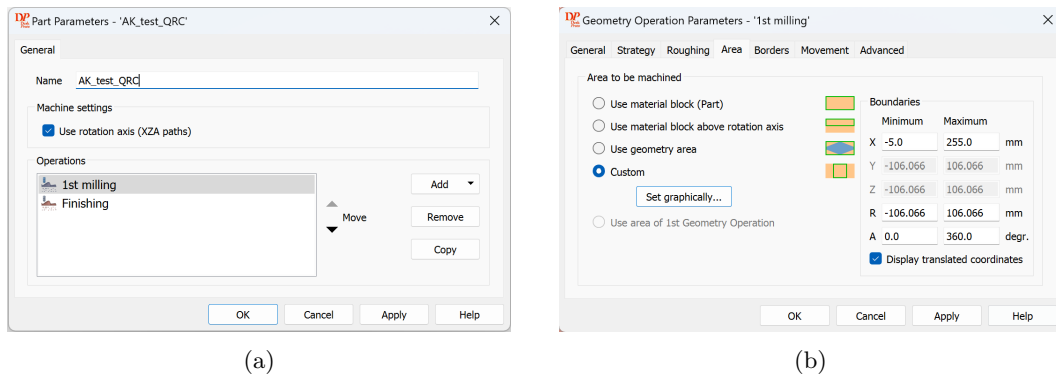


Figura 5.24: Configurações gerais dos parâmetros da peça (a) e das dimensões do bloco a maquinar (b), no *software* DeskProto

Após serem definidos os parâmetros relativos à peça a maquinar, têm de ser configurados os parâmetros para as operações de maquinagem, sendo que, habitualmente, são criadas duas operações distintas, *first milling* (fresagem inicial) e *finishing* (acabamento), ao ser importante que o molde positivo seja obtido com a melhor qualidade e definição possíveis. Estes parâmetros são definidos segundo as características do material a ser fresado, bem como as características do *spindle*.

A primeira operação realizada é uma fresagem mais robusta, e a segunda operação, é uma fresagem com mais detalhe, levando a um acabamento da peça mais minucioso e com melhor acabamento. Em qualquer uma das situações, a primeira coisa a definir é a ferramenta de corte, que neste caso será a mesma, e, além disso, são definidos parâmetros de precisão e velocidade da ferramenta de corte e *spindle*, a estratégia de fresagem e a modo como esta ocorre, em cada uma das operações.

Em seguida são mostradas as configurações necessárias em cada uma das operações, conforme o resultado pretendido nas mesmas:

1. *First milling*: em primeiro lugar, definem-se os valores de precisão da ferramenta de corte e a velocidade de atuação do *spindle*. Como se pode verificar na **Figura 5.25a**, é definida uma distância entre *paths* de 10 mm, pois como se trata da primeira fresagem poderá ser utilizada esta distância entre *paths*, e definido um passo entre *paths* de 0,667 mm para o caso de não ocorrer uma profundidade igual ao diâmetro máximo da ferramenta, evitar o aparecimento de zonas onde o material não é fresado completamente. O parâmetro da velocidade de avanço é definido para 700 mm/min e a velocidade do *spindle* para 2000 rpm como se observa na **Figura 5.25a**.

A seguir, é definida a estratégia de fresagem para esta operação, sendo definido que a mesma acontece paralelamente ao eixo A, de rotação, visível na **Figura**

5.25b, uma vez que o molde positivo que se pretende obter tem uma forma cilíndrica, logo a melhor estratégia de fresagem é a revolução.

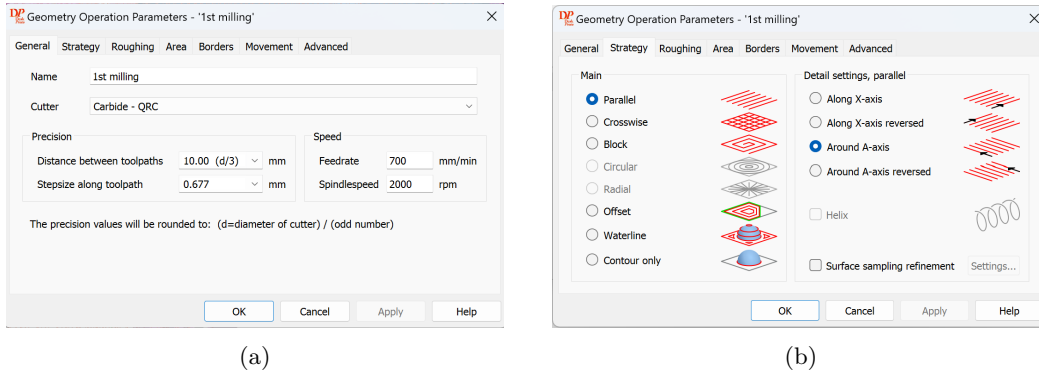


Figura 5.25: Configuração dos valores de precisão e velocidade (a) e estratégia de fresagem (b) para a primeira operação, no *software* DeskProto

Por fim, como se observa na **Figura 5.26a**, define-se a profundidade que a ferramenta alcança em cada fresagem, definindo para 5 mm e deixando uma margem de 10 mm de material em todo o volume para, posteriormente, ser realizada a fresagem de acabamento. Além disto, também é configurado o movimento que a ferramenta de corte realiza ao remover o material do bloco, como se visualiza na **Figura 5.26b**, definindo-o para *Meander* facilitando a fresagem ao promover a remoção do material sem ser necessário ser afastada e reposicionada com frequência, diminuindo o tempo de fresagem porque desta maneira a ferramenta também se move continuamente de 0 para 360° e vice-versa, a cada passo. Ainda, é definido o afastamento da ferramenta à peça quando esta tem de se movimentar livremente, havendo uma distância de 5 mm entre o bloco e a ferramenta.

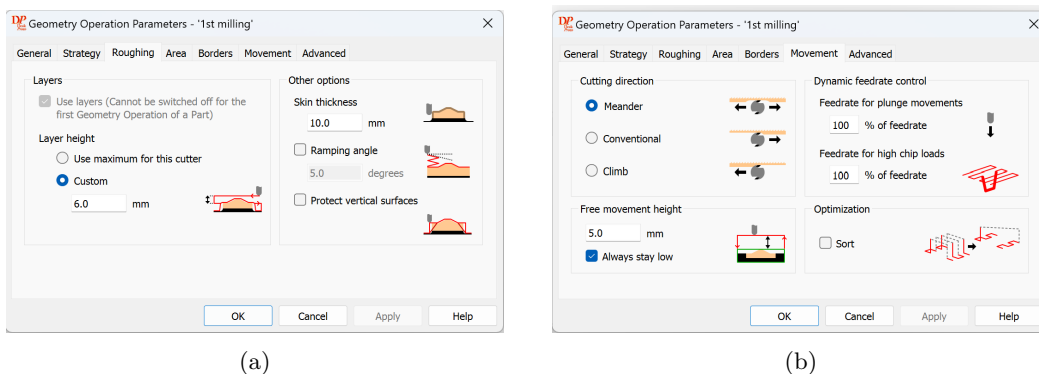


Figura 5.26: Configuração de corte e profundidade (a) e movimento (b) da ferramenta, para a primeira operação, no *software* DeskProto

2. *Finishing*: tal como na primeira operação, inicialmente, definem-se os valores de precisão da ferramenta de corte e a velocidade de atuação do *spindle*. Como se pode verificar na **Figura 5.27**, é definida uma distância entre *paths* de 2 mm e um passo entre *paths* de 0,08 mm, pelas mesmas razões mencionadas para a primeira operação, mas com valores inferiores, para que o resultado da fresagem tenha um acabamento com uma boa qualidade. O parâmetro da velocidade de avanço é definido para 700 mm/min e a velocidade do *spindle* para 1200 rpm como se observa na **Figura 5.27**.

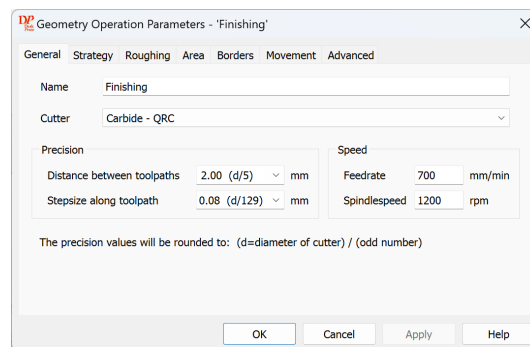


Figura 5.27: Configuração dos valores de precisão e velocidade, para a segunda operação, no *software* DeskProto

A seguir, é definida a estratégia de fresagem para esta operação, sendo a mesma utilizada para a primeira operação.

Por fim, como se observa na **Figura 5.28**, define-se a profundidade que a ferramenta alcança em cada fresagem, definindo para a fresagem ser realizada até à obtenção do molde positivo, passando a margem deixada de 10 mm para 0, que corresponde ao acabamento. Tal como na primeira operação, o movimento que a ferramenta de corte realiza ao remover o material do bloco mantém-se, bem como a distância entre o bloco e a ferramenta quando ocorre movimento livre.

De realçar que a escolha dos parâmetros de maquinagem, tais como a velocidade de corte ou até mesmo a distância entre *paths* depende de vários fatores nomeadamente o tipo de material a ser maquinado ou do acabamento que se pretende obter na peça. A estratégia de maquinagem é ajustada com base no equilíbrio entre a eficiência e precisão, com a fase inicial sendo focada na remoção rápida do material e a fase final priorizando a qualidade da superfície.

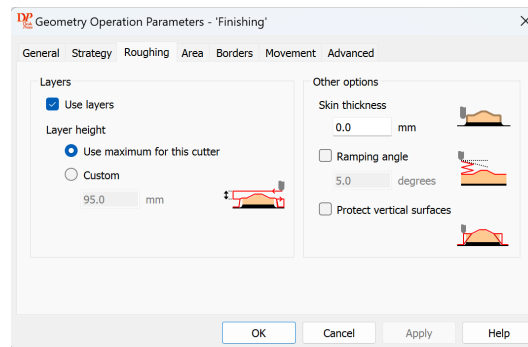


Figura 5.28: Configuração de corte e profundidade da ferramenta, para a segunda operação, no *software* DeskProto

Simulação do tempo de maquinagem e da peça final

Depois de serem inseridos os parâmetros necessários para as duas operações descritas acima, o *software* gera os *paths* que a ferramenta de corte realiza em cada uma das operações e simula o tempo que a máquina demora a realizar a fresagem da peça, observado nas **Figuras** 5.29 e 5.30.

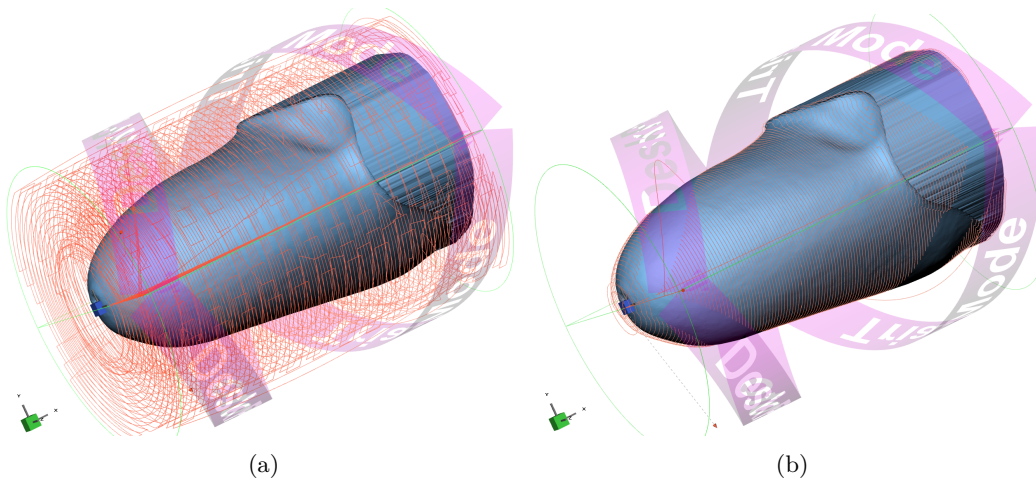


Figura 5.29: Simulação dos *paths* que a ferramenta de corte realiza na operação *first milling* (a) e na operação *finishing* (b), no *software* DeskProto

Observando o tempo estimado para as duas operações, verifica-se que se trata de um valor elevado que poderá estar associado a alguns fatores, que serão discutidos posteriormente, no **Capítulo** 6.

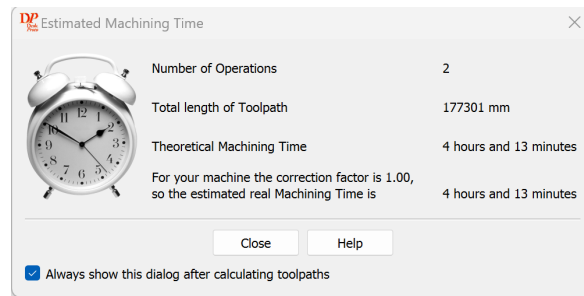


Figura 5.30: Estimação do tempo de fresagem para as duas operações, no *software* DeskProto

Antes de ser gerado o G-code, o *software* simula o formato do bloco a ser maquinado, bem como o molde positivo obtido após a maquinagem do mesmo. Estas simulações podem ser analisadas na **Figura 5.31**.

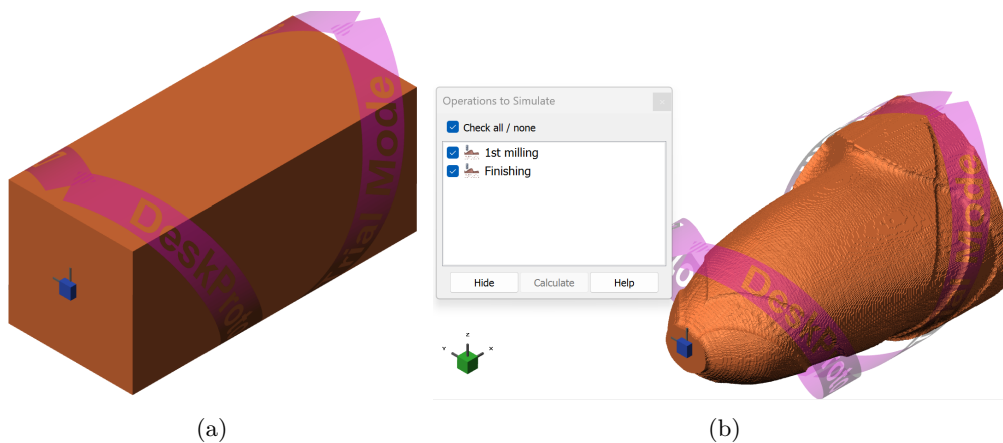


Figura 5.31: Simulação do bloco de PU (a) e do molde positivo (b), no *software* DeskProto

Exportar o G-code

Para concluir o processo é gerado o G-code, visível na **Figura 5.32**, que é exportado, seguidamente, no formato `.nc`, para poder ser utilizado na máquina CNC. É possível observar as diferenças nas partes iniciais do G-code, onde na **Figura 5.32a** observa-se que a deslocação no eixo X ocorre a cada 10 mm, enquanto na **Figura 5.32b** a deslocação acontece a cada 2 mm.

1	%	21676	G00 X-5.000 A0.000 S1200 M3
2	G90	21677	G01 Z11.066 F700
3	G00 X-5.000 Z111.066 A25.200 S2000 M3	21678	G01 A86.177
4	G01 Z101.066 F700	21679	G01 Z11.085 A86.427
5	G01 A64.800	21680	G01 Z11.138 A87.091
6	G01 X0.000	21681	G01 Z11.194 A87.756
7	G01 A25.200	21682	G01 Z11.315 A89.086
8	G01 X10.000	21683	G01 Z11.445 A90.416
9	G01 A64.800	21684	G01 Z11.587 A91.745
10	G01 X20.000	21685	G01 Z11.656 A92.410
11	G01 A25.200	21686	G01 Z11.721 A93.158
12	G01 X30.000	⋮	
13	G01 A64.800	21766	G01 Z11.066 A171.856
14	G01 X40.000	21767	G01 A360.000
15	G01 A25.200	21768	G01 X-4.000
16	G01 X50.000	21769	G01 A171.856
17	G01 A64.800	21770	G01 Z11.069 A171.773
18	G01 X60.000	⋮	
19	G01 A25.200	21852	G01 Z11.066 A86.177
20	G01 X70.000	21853	G01 A0.000
21	G01 A64.800	21854	G01 X-2.000
22	G01 X80.000	21855	G01 A39.141
23	G01 A25.200	21856	G01 Z11.496 A39.889
24	G01 X90.000	⋮	
25	G01 A64.800	22147	G01 Z11.066 A289.030
26	G01 X100.000	22148	G01 A360.000
27	G01 A25.200	22149	G01 X0.000
28	G01 X110.000	22150	G01 A318.615
29	G01 A64.800	22151	G01 Z12.715 A317.950
30	G01 X120.000		

Figura 5.32: Partes iniciais do G-code da operação *first milling* (a) e *finishing* (b)

5.3.4 Preparar bloco de PU para maquinar

A maquinação do molde positivo implica a preparação do bloco de PU, com as dimensões necessárias para a reprodução do molde 3D. A empresa compra blocos de PU, de várias densidades, e com um volume de 2000*800*700 mm, cortando-os consoante a medida necessária para a maquinação da peça. Por questões práticas e de aproveitamento de material, foi utilizado um bloco com a densidade de 55 kg/m³.

Depois de definidas as dimensões necessárias do bloco, conforme os valores inseridos no *software* DeskProto, observado na **Figura 5.24**, o mesmo tem de ser fixado à peça de fixação da máquina, visível na **Figura 5.33a**, através de parafusos que fixam, em segurança, o bloco à peça de fixação, minimizando qualquer tipo de vibrações indesejadas. A peça de fixação é, posteriormente, acoplada ao eixo A, de rotação, como se pode observar pela **Figura 5.33b**.

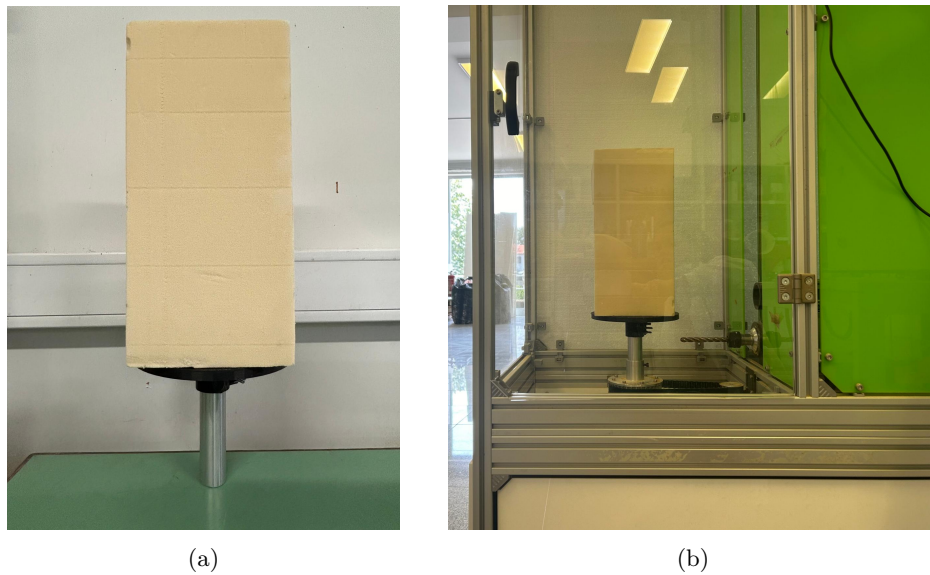


Figura 5.33: Bloco de PU fixado à peça de fixação (a) e colocado na máquina CNC no eixo A (b)

5.3.5 Preparar máquina para a maquinaagem

Quando estão definidos todos os passos enumerados nos **Capítulos** 5.3.1 a 5.3.4, é fundamental preparar a máquina CNC para que a maquinaagem aconteça.

Em primeiro lugar é essencial ligar a fonte de alimentação da máquina à corrente. Em seguida, ativar a passagem de corrente para o sistema motor e para o controlador e, por fim, é necessário desativar o travão dos motores. Estas tarefas podem ser conseguidas através dos botões sinalizados na caixa de controlo, observáveis na **Figura 5.34**

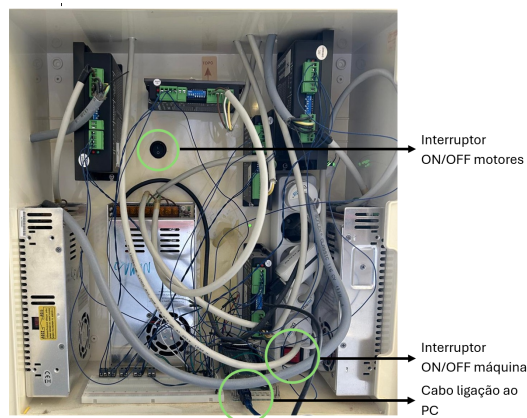


Figura 5.34: Interruptores ON/OFF da máquina "QRC"

Após a máquina se encontrar ligada, é necessário ser estabelecida a ligação do controlador à mesma. Para tal, é utilizado um cabo *Universal Serial Bus* (USB)

2.0, como se observa na **Figura 5.34**, ligando um PC à máquina, pois o controlador utilizado, o Grbl *Controller*, só o consegue fazer através deste meio de comunicação.

Como referido no **Subcapítulo 4.1.3**, o Grbl, é o *software* utilizado para realizar o controlo da máquina, pelo que, antes de qualquer operação de maquinagem ser realizada, a porta onde se insere o controlador Arduino tem de estar ativada e seguidamente o Grbl arranca, automaticamente, o canal da máquina.

Ainda antes do bloco a maquinar ser colocado na máquina, e como referido anteriormente, no **Subcapítulo 5.3.3**, é necessário ser definido o zero do eixo Z, manualmente, uma vez que este se encontra no centro do eixo A, como se pode ver na **Figura 5.20**. Assim, para isto ocorrer, têm de ser enviados comandos de G-code, em coordenadas relativas (G91), pela linha de comandos, do *software*, sendo importante analisar quando a extremidade da ferramenta de corte atinge o centro do eixo A, pois este processo é feito manualmente. Depois da ferramenta se encontrar no centro do eixo Z é enviado o seguinte comando, que define o zero do eixo Z:

```
1 G92 Z0
```

Além de ser definido o zero do eixo Z, também é importante fazer esta operação para o eixo X, mas, contrariamente à operação realizada para o eixo Z, o bloco de material a maquinar, tem de ser colocado na mesa de trabalho, como exemplificado na **Figura 5.33b**. Deste modo, é essencial, inserir, uma vez mais, na linha de comandos, G-code idêntico ao inserido para o eixo Z, substituindo Z por X, tendo em consideração a altura dos parafusos que fixam o bloco à peça de fixação e uma margem de bloco para ser possível a peça ser maquinada completamente. Depois da ferramenta se encontrar no centro do eixo X é enviado o seguinte comando, que define o zero do eixo X:

```
1 G92 X0
```

Depois de serem realizadas as parametrizações dos zeros do eixo Z e X, é possível enviar, pela linha de comandos, os comandos do G-code em coordenadas absolutas (G90) para a maquinagem do molde positivo.

Pela **Figura 5.35** consegue-se verificar a posição inicial que a ferramenta de corte vai adotar após a inserção do G-code no *software*.

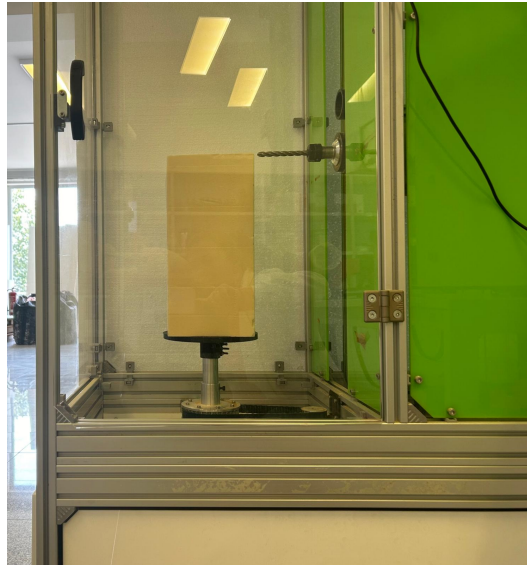


Figura 5.35: Posição inicial da ferramenta de corte aquando da ma-quinagem do bloco de PU

Para concluir o processo de preparação da máquina, antes da maquinação da peça, o *spindle* tem de ser ligado, como se pode ver na **Figura 5.36**, para o processo de maquinação poder acontecer.



Figura 5.36: Controlador do *spindle*

A velocidade de operação do *spindle* não pode ser definida diretamente, sendo necessário inserir a gama de frequência a que o mesmo opera, conforme o tipo de material que será maquinação. Para o caso em estudo, é utilizada uma frequência entre 85 e 90 Hz.

5.3.6 Maquinagem do bloco de PU

A maquinagem do bloco de PU é o processo final do fabrico do molde positivo, uma vez que é nesta fase que é fresado o bloco de material em bruto. Depois de serem realizados os passos descritos no **Subcapítulo** 5.3, como a configuração do G-code e a fixação do bloco na máquina, é iniciado o processo de maquinagem.

Assim, o processo inicia-se com o acionamento do *spindle*, que gira à velocidade determinada pela frequência ajustada, como referido no **Subcapítulo** 5.3.5, de 85 a 90 Hz. A ferramenta de corte, acoplada ao *spindle*, vai mover-se ao longo dos eixos X e Z, enquanto o eixo A realiza a rotação contínua do bloco de PU.

Como foi possível observar no **Subcapítulo** 5.3.3, pela parametrização da maquinagem no DeskProto, habitualmente, o processo de maquinagem encontra-se dividido em duas etapas:

1. Fresagem inicial: nesta fase, a ferramenta de corte remove o material que se encontra em excesso, aproximando-se da forma final do molde, deixando margem para a fase final. A estratégia de fresagem revolutiva, onde a ferramenta percorre o comprimento do bloco enquanto o eixo A gira, permite uma remoção eficiente e controlada do material.
2. Acabamento: depois de ser efetuada a fresagem inicial, realiza-se uma operação de acabamento para ocorrer uma fresagem da superfície mais detalhada, corrigindo imperfeições deixadas pela fresagem inicial. A ferramenta de corte faz movimentos mais precisos e detalhados para atingir a forma exata.

Durante o processo de maquinagem, é crucial ser monitorizado a profundidade de corte e a distância entre os *paths* da ferramenta para evitar falhas na maquinagem, como vibrações excessivas ou remoção incompleta do material. O resultado da maquinagem do bloco de PU é um molde positivo pronto para ser utilizado na fabricação de dispositivos médicos.

Em seguida, no **Subcapítulo** 5.4 são mostrados os resultados obtidos com a maquinagem do bloco de PU pelos parâmetros definidos previamente.

5.4 Resultados obtidos com a maquinagem

Depois de finalizado o processo de maquinagem do bloco de PU com o intuito de se obter um molde positivo para o fabrico de uma prótese para o membro inferior, é possível observar pela **Figura** 5.37 que se conseguiu obter o mesmo.

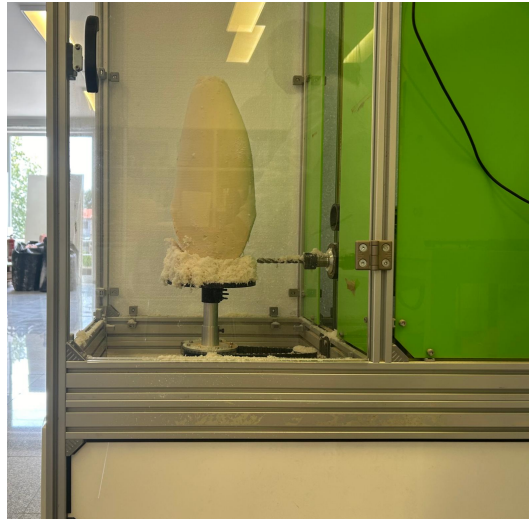


Figura 5.37: Bloco de PU maquinado

O processo de maquinagem, apesar de se encontrar estimado para 4h e 13 min excedeu este tempo, verificando-se que o bloco foi maquinado em 5h. Apesar do processo demorado conseguiu obter-se um molde positivo com um acabamento satisfatório, notando-se algumas linhas de transição entre *paths* que poderiam ser minimizadas reduzindo a distância entre as mesmas.

Durante o processo de maquinagem foram observados alguns aspetos que podem ser melhorados posteriormente:

- O *feedrate* poderia ser maior uma vez que a vibração sentida na máquina derivada dos motores não foi excessiva;
- A velocidade de rotação dos motores poderia igualmente ter sido maior pelos motivos mencionados;
- A utilização de um bloco cilíndrico poderia ter reduzido o tempo de maquinagem uma vez que seria necessário a remoção de menor material durante a maquinagem do mesmo;
- O aumento do tempo de maquinagem face ao simulado pode ter sido derivado de um atraso no processamento do G-code.

Os aspetos verificados durante a maquinagem podem estar relacionados pela falta de realização de testes de precisão e repetibilidade após a configuração dos parâmetros inseridos no Grbl, referidos no **Subcapítulo 5.2**. Com estes testes poderia se chegar à conclusão que era necessário ajustar os valores inseridos para ir de encontro a um processo de maquinagem mais adequado e menos moroso.

5.5 Limitações encontradas na máquina "QRC"

A máquina CNC "QRC", apesar de cumprir o objetivo que é o fabrico de moldes positivos, apresenta algumas limitações que podem influenciar o seu desempenho em algumas situações.

Pelas dimensões da máquina, é possível perceber que não é possível serem obtidos moldes positivos com grande volume e dimensão, como moldes para o fabrico de ortótese para a coluna. Para isto ser possível seria necessário proceder-se a uma alteração na sua estrutura permitindo a maquinação de blocos de maior dimensão.

Além disso, a máquina encontra-se desenhada para maquinação, essencialmente, materiais de baixa a média densidade, como o PU, dificultando a utilização de outro tipo de material com características de densidade maiores ou mais reduzidas.

Outra limitação percebida foi que apesar da máquina oferecer a precisão suficiente para o fabrico dos moldes, as vibrações dos motores, os ajustes manuais dos parâmetros do *spindle* e a estrutura física da máquina, podem provocar pequenas variações nos moldes, particularmente quando as operações são mais complexas e se pretende obter uma elevada precisão.

Como referido no parágrafo anterior, o ajuste manual do *spindle* limita a automatização dos processos de maquinação, ao impedir que sejam realizados ajustes automáticos durante o processo. Uma automatização deste sistema poderia melhorar o desempenho da maquinação quando são necessárias operações mais precisas e complexas.

Por fim, o tempo de maquinação do bloco é relativamente longo, quando comparado com outras máquinas CNC existentes no mercado. Este fator pode estar associado ao sistema de transmissão de movimento que como referido no **Subcapítulo 4.1** é mais económico e acessível, o que por vezes pode diminuir o seu desempenho, tendo em conta a área onde é aplicado.

As limitações apresentadas sugerem potenciais melhorias no sistema o que pode aumentar a aplicabilidade da máquina.

5.6 Plano de manutenção e melhorias futuras para o sistema

Quando se procedeu à análise e estudo da máquina CNC percebeu-se que a mesma apresentava alguns pontos que poderiam ser melhorados. Um destes pontos prende-se com o facto de não ser realizada aspiração do material desperdiçado aquando da maquinação dos blocos de PU. Além disso, percebeu-se que não são realizadas manutenções periódicas nem após as maquinações, pelo que seria importante estruturar um plano de manutenção, para prevenir problemas e assegurar a qualidade e precisão das operações de maquinação.

Como tal, abaixo encontram-se alguns aspetos que devem ser considerados para esta situação:

1. Manutenção regular:

- Verificação periódica das ligações elétricas, nomeadamente das ligações entre as fontes de alimentação e as ligações de comunicação entre os diferentes componentes da máquina, como *driver*, motores e controlador;
- Manutenção preventiva dos motores, com a lubrificação periódica das partes móveis dos motores, como os fusos e os rolamentos, para garantir um movimento suave e reduzir o seu desgaste;
- Lubrificação dos eixos e guias de acordo com as especificações dos fabricantes destes componentes de modo a reduzir o atrito e o risco de possíveis travamentos, facilitando a precisão dos movimentos;
- Inspeção do funcionamento do *spindle* para perceber se encontra dentro dos parâmetros corretos de rotação e torque. Além disso, monitorizar sinais de desgaste, ruídos anormais ou sobrecargas, que podem indicar a necessidade de manutenção ou substituição de componentes internos;
- Monitorização da temperatura dos *drivers* e dos motores durante a operação, uma vez que temperaturas elevadas podem indicar problemas de ventilação ou sobrecarga, levando ao sobreaquecimento e falhas nos componentes;
- Testes de comunicação antes de cada processo de maquinagem, através de ciclos de teste, para perceber se o G-code enviado está a ser processado adequadamente sem haver problemas de comunicação entre o PC, controlador e a máquina;
- Ajustes no *software* através da realização de calibrações periódicas aos eixos com o intuito de manter a precisão da máquina. Estas calibrações garantem que a máquina realiza os movimentos esperados pelo envio do G-code e mantem a precisão dimensional das peças maquinadas.

2. Melhorias a implementar ao sistema CNC:

- Automatização do *spindle*, implementando, por exemplo, uma chave automática, como se verifica em máquinas CNC industriais, que liga e ajusta a velocidade de acordo com as necessidades de maquinagem. Isto permite um controlo mais eficiente deste componentes durante as operações de maquinagem e reduz a intervenção humana o que aumenta a precisão e a segurança durante este processo;

- Incluir sensores de fim de curso em todos os eixos uma vez que auxiliam na prevenção de colisões com a peça a maquinar e garantem que os movimentos da máquina se encontram nos limites físicos da mesma. Além disso, a inclusão de sensores pode promover o processo de *homing* que não se encontra definido neste sistema;
- Incluir mais eixos de movimento, através da adição de um ou mais eixos, o que pode aumentar a complexidade e flexibilidade das operações de maquinagem. Este fator pode facilitar a maquinagem de peças mais complexas e com geometrias mais desafiantes.

5.7 Conclusão do Capítulo 5

Neste capítulo, foi apresentada a implementação do sistema CNC, destacando as etapas essenciais para a integração entre o *hardware* e o *software*, com a utilização do Arduino Mega 2560 e do *firmware* Grbl-Mega-5X. Foram detalhadas as ligações entre os motores, *drivers* e fontes de alimentação, enfatizando a importância de garantir ligações estáveis e ajustadas para o correto funcionamento do sistema.

Além disso, foi realizada uma integração eficaz entre o controlador Arduino e o Grbl Controller, permitindo a comunicação adequada entre o PC e a máquina CNC, etapa que permitiu o envio de comandos G-code, resultando no processo de maquinagem do molde positivo. A implementação foi concluída com sucesso, apesar dos desafios enfrentados nas ligações e nos ajustes dos parâmetros do sistema.

A análise dos resultados práticos, bem como a identificação de limitações e sugestões de melhoria, são abordadas no **Capítulo 6**, com o intuito de otimizar o desempenho da máquina CNC e garantir maior precisão e eficiência nos processos de maquinagem.

Capítulo 6

Conclusões

Neste capítulo são apresentadas as conclusões finais relativas ao trabalho realizado, as principais dificuldades sentidas durante a realização do mesmo e, ainda, ideias para trabalhos futuros e melhorias no projeto.

6.1 Considerações finais

O desenvolvimento e a requalificação de uma máquina CNC de três eixos, aplicada ao fabrico de moldes positivos para próteses e ortóteses, foi o principal objetivo desta dissertação. Foram implementadas diversas melhorias em nível de *hardware* e *software*, abrangendo desde a reconfiguração das ligações elétricas e dos *drivers* até à adaptação do *firmware* Grbl-Mega-5X para controlar a máquina.

Na fase inicial do projeto, realizou-se uma análise comparativa entre o fabrico tradicional e digital dos moldes em PU na área da ortoprotesia, estudando detalhadamente os componentes essenciais das máquinas CNC e as tecnologias disponíveis no mercado. A partir desta pesquisa, foi possível identificar as limitações existentes no atual sistema, como a complexidade na configuração dos parâmetros de maquinagem e a integração inadequada dos componentes.

Os ajustes às ligações cabladas e a reconfiguração do mapa de pinos entre o *firmware* e o controlador foram etapas críticas para assegurar o correto funcionamento dos motores e dos eixos. A integração do *firmware* Grbl-Mega-5X, juntamente com o controlador Arduino Mega 2560, proporcionou um controlo mais flexível e a

possibilidade de expansão da máquina até cinco eixos. A interface gráfica desenvolvida no Grbl Controller facilitou a operação da máquina, permitindo a programação e o monitorização dos processos de maquinagem de maneira mais intuitiva e precisa.

Embora tenham sido enfrentados alguns desafios, como a adaptação dos parâmetros de controle para materiais específicos e a necessidade de reconfiguração dos motores, foi possível realizar a maquinagem de um molde em PU. O resultado obtido foi satisfatório em termos de qualidade, mas o tempo de processamento ainda se mostrou elevado, possivelmente devido à configuração dos parâmetros de maquinagem, como a velocidade de avanço e a profundidade de corte. Futuramente, a realização de testes adicionais é essencial para ajustar os parâmetros e otimizar o processo.

Para melhorar o desempenho da máquina, foi elaborado um plano de manutenção preventiva que inclui a verificação periódica das ligações elétricas, a lubrificação das partes móveis e a calibração regular dos eixos. Estas práticas visam garantir o bom funcionamento e a precisão dos componentes, além de aumentar a longevidade do sistema e prevenir falhas inesperadas.

No entanto, ainda existem limitações na máquina que precisam de merecem atenção no futuro. A adaptação para maquinagem de outros materiais, a integração de sensores de fim de curso e segurança e a automatização completa do *spindle* são melhorias que podem aumentar significativamente a eficiência e a versatilidade da máquina. Além disso, a criação de uma interface gráfica portátil poderia proporcionar maior flexibilidade aos operadores, facilitando o controlo da máquina em diferentes ambientes de produção.

Em suma, este trabalho demonstrou a viabilidade e os benefícios de requalificar uma máquina CNC utilizando tecnologias acessíveis e adaptáveis. As melhorias implementadas aumentam a precisão e a qualidade dos moldes produzidos, mas também estabelecem uma base sólida para futuras expansões e otimizações do sistema. Com as melhorias sugeridas, espera-se que a máquina se torne uma ferramenta ainda mais eficaz e versátil na fabricação de dispositivos médicos personalizados.

6.2 Trabalho Futuro

Após a conclusão deste projeto, pode-se afirmar que, apesar de nem todos os objetivos terem sido atingidos, a máquina CNC "QRC" encontra-se funcional. No entanto, existem algumas melhorias que podem ser implementadas no futuro para otimizar o desempenho da máquina e ampliar a sua aplicabilidade. Entre as principais melhorias destacam-se as seguintes:

- **Automatização do *spindle*:** o ajuste atual do *spindle* é realizado manualmente, o que limita a automação completa do processo de maquinagem. A integração de um sistema de controlo automático para este componente permitiria que os

parâmetros de velocidade fossem ajustados dinamicamente durante a operação, de acordo com alguns fatores como a dureza do material ou as condições da ferramenta. Esta melhoria poderia resultar numa operação mais eficiente e precisa, otimizando o tempo de produção e a qualidade final dos moldes;

- Utilização de materiais mais resistentes na estrutura da máquina: o foco inicial da máquina era a fresagem de PU, material de baixa a média densidade. No entanto, durante o processo de maquinagem, foram observadas vibrações que poderiam ser minimizadas caso a estrutura fosse mais robusta. Além disso, e com o objetivo de poderem ser maquinados outro tipo de materiais, seria adequado fazer uma reestruturação da máquina, incluindo motores mais potentes e uma estrutura mais robusta. Estas modificações poderiam aumentar a versatilidade da máquina em diferentes tipos de aplicações;
- Aumento da área de trabalho: como referido, as dimensões da máquina e a área de trabalho da mesma, são de tamanho reduzido, quando se compara com máquinas existentes no mercado com o mesmo objetivo desta. A adaptação para trabalhar com blocos de maiores dimensões, através do aumento do comprimento dos eixos ou da maior rotação do eixo A, poderia expandir a capacidade da mesma, o que seria particularmente útil para a fabricação de moldes de dispositivos ortopédicos maiores, como ortóteses para a coluna. Além disso, seria igualmente vantajoso a adição de um eixo de movimento para produzir outro tipo de produtos, que requerem formas ainda mais detalhadas, como, por exemplo, palmilhas ortopédicas;
- Diminuição do tempo de maquinagem: durante os testes, observou-se que o tempo necessário para completar um ciclo de maquinagem era relativamente longo, o que torna demorado todo o processo de fabrico de próteses e ortótese. A implementação de sistemas de transmissão de movimento mais eficientes, utilizando motores mais rápidos ou técnicas mais avançadas de transmissão, como utilização de outro tipo de *firmware*, poderia reduzir significativamente o tempo total de produção sem comprometer a precisão;
- Interface do utilizador portátil: apesar de ter sido implementada uma interface gráfica através do PC, a utilização de uma interface portátil seria ainda mais vantajosa. Isto permitiria ao operador controlar a máquina a partir de diferentes locais na empresa, sem a necessidade de estar fisicamente junto à mesma. Além disso, uma interface portátil ofereceria maior ergonomia, evitando que o operador precise trabalhar em posições desconfortáveis ou potencialmente perigosas.

- sistema de aspiração: a máquina CNC não apresenta, atualmente, um sistema de aspiração contínuo, sendo a mesma apenas limpa, com um aspirador industrial, após cada maquinagem. A implementação de um sistema de aspiração dedicado traria melhorias significativas à operação da máquina, mantendo o ambiente de trabalho limpo e organizado, protegendo os componentes mecânicos da máquina e melhorando a qualidade das peças produzidas. Isso também otimizaria a eficiência do processo de maquinagem.

Estas melhorias futuras aumentariam a aplicabilidade da máquina numa variedade maior de operações e materiais, além de proporcionar um aumento significativo na produtividade e eficiência da produção de moldes ortopédicos.

Referências

- [1] Eubomír Šooš and J. Marek, *Machine Tools - Design, Research, Application*. Rijeka: IntechOpen, Oct 2020. [Citado nas páginas xii, 3 e 33]
- [2] D. G. Shurr and J. W. Michael, *Prosthetics & Orthotics*. Prentice Hall, second edition ed., 2002. [Citado nas páginas xvii, 8, 9, 11 e 15]
- [3] S. Hussain, S. Shams, and S. J. Khan, “Impact of medical advancement: Prostheses,” in *Computer Architecture in Industrial, Biomechanical and Biomedical Engineering* (L. Wang and L. Yu, eds.), ch. 2, Rijeka: IntechOpen, 2019. [Citado nas páginas xi, xvii, 8, 10 e 13]
- [4] A. Heaver, P. Kraft, G. Morell, A. Platzer, Ralph Roloff, and W. F. Sauter, *Orthotics and Prosthetics Digest - Reference Manual (Second Edition)*. Edahl Procutions Ltd., second edition ed., 1983. [Citado nas páginas xvii, 9, 10, 11, 12, 13 e 14]
- [5] G. Fitzlaff and S. Heim, *Lower Limb Prosthetic Components : Design, Function and Biomechanical Properties*. Verlag Orthopadie-Technik, 2002. [Citado nas páginas xi, 9 e 10]
- [6] M. M. Lusardi and C. C. Nielsen, *Orthotics and prosthetics in rehabilitation*. Butterworth Heinemann, 2000. [Citado nas páginas xi, 11, 13, 24 e 26]
- [7] J. Barrios-Muriel, F. Romero-Sánchez, F. J. Alonso-Sánchez, and D. R. Salgado, “Advances in orthotic and prosthetic manufacturing: A technology review,” *Materials*, vol. 13, 2020. [Citado nas páginas xi, 11 e 12]
- [8] M. K. Tiwari and N. Tiwari, “Lower limb orthotics: An overview,” *International Journal of Health Sciences and Research*, vol. 12, no. 10, 2022. [Citado nas páginas xi e 11]
- [9] D. G. Pitts and E. E. Fess, *7 - Orthoses: Essential Concepts*. St. Louis: Mosby, second edition ed., 2014. [Citado nas páginas xi e 12]
- [10] A. Ali, V. Fontanari, M. Fontana, and W. Schmölz, “Spinal deformities and advancement in corrective orthoses,” *Bioengineering*, vol. 8, no. 1, 2021. [Citado nas páginas xi e 12]

- [11] G. F. Kogler, M. Bridges, J. E. Hilliard, and K. K. Chui, *Orthotics and Prosthetics in Rehabilitation (Fourth Edition)*. Elsevier, fourth edition ed., 2020. [Citado nas páginas xi, 14, 15, 17, 18, 20, 22, 23 e 24]
- [12] “Ligadura gessada.” Available at <https://www.bastosviegas.com/ligaduras-elasticas/ligadura-gessada>. (Last accessed in 10/04/2024). [Citado nas páginas xi e 15]
- [13] “Ligadura gessada.” Available at <https://www.factormed.pt/pt/catalogo/ligaduras/gessadas/ligadura-sintetica/>. (Last accessed in 10/04/2024). [Citado nas páginas xi e 15]
- [14] D. R. F. for the Orthotics & Prosthetics Community, “Lower-extremity orthotics measurement form.” Available at <https://www.oandplibrary.org/popup.asp?frmItemId=298DDFB7-59D0-42ED-9FEC-1D5DAC1822B4&frmType=image&frmId=4>. (Last accessed in 11/04/2024). [Citado nas páginas xi e 16]
- [15] “3d patient scans for orthotics and prosthetics of all types.” Available at <https://vorum.com/spectra-3d-scanner/>. (Last accessed in 23/04/2024). [Citado nas páginas xi, 19 e 21]
- [16] “Ortenshape - all-in-one software to design your orthopedic device.” Available at <https://orten.proteor.com/ortenshape>. (Last accessed in 23/04/2024). [Citado nas páginas xi, xii, 19 e 58]
- [17] J. Barrios-Muriel, F. Romero-Sánchez, F. J. Alonso-Sánchez, and D. R. Salgado, “Advances in orthotic and prosthetic manufacturing: A technology review,” *Materials*, vol. 13, 1 2020. [Citado na página 20]
- [18] S. Grazioso, M. Selvaggio, T. Caporaso, and G. D. Gironimo, “A digital photogrammetric method to enhance the fabrication of custom-made spinal orthoses,” *Journal of Prosthetics and Orthotics*, vol. 31, no. 2, 2019. [Citado nas páginas 21, 22 e 23]
- [19] G. Nehme and S. Ghalambor, “Optimization of important relief areas in prosthetic socket for below-knee amputees using design of experiment and finite element model,” *Journal of Prosthetics and Orthotics*, vol. 26, no. 4, 2014. [Citado nas páginas xi e 22]
- [20] M. Geoffroy, J. Gardan, J. Goodnough, and J. Mattie, “Case report cranial remodeling orthosis for infantile plagiocephaly created through a 3d scan, topological optimization, and 3d printing process,” *Journal of Prosthetics and Orthotics*, vol. 30, no. 4, 2018. [Citado nas páginas xi e 23]

- [21] S. Kumar, S. Bhowmik, and V. K. Mahakur, “Thermo-mechanical and degradation properties of naturally derived biocomposites for prosthesis applications: Analysis of the interface pressure and stress distribution on the developed socket,” *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, vol. 237, pp. 1999–2015, 10 2023. [Citado nas páginas xi e 25]
- [22] A. Vitali, D. Regazzoni, C. Rizzi, and G. Colombo, “Design and additive manufacturing of lower limb prosthetic socket,” 2017. [Citado nas páginas xi e 25]
- [23] S. H. Suh, S. K. Kang, D. hyuk Chung, and I. A. Stroud, *Theory and Design of CNC Systems*. Springer, 2008. [Citado nas páginas xii, xvii, 28, 30, 32, 34, 35, 36, 37, 39, 48, 50, 51, 52, 53 e 54]
- [24] M. P. Groover, *Automation, Production Systems, and Computer-Integrated Manufacturing*. Prentice Hall PTR, 2nd ed., 2000. [Citado nas páginas xii e 30]
- [25] M. Dean, “Cnc milling coordinate system made easy.” Available at www.autodesk.com/products/fusion-360/blog/cnc-coordinate-system-made-easy/. (Last accessed in 9/08/2024). [Citado nas páginas xii, 30 e 31]
- [26] “Código g para cnc: o guia completo.” Available at <https://www.tsinfa.com/pt/g-code-for-cnc/>. (Last accessed in 12/08/2024). [Citado nas páginas xvii e 32]
- [27] H. Youssef and H. El-Hofy, *Introduction to machine and machine tools*. Taylor & Francis Group, second ed., 2021. [Citado nas páginas 32, 33, 34, 43, 71, 72, 73 e 74]
- [28] I. G. Torres, “Estudo de requalificação de máquinas-ferramenta cnc,” Master’s thesis, Universidade do Minho - Escola de Engenharia, 2021. [Citado nas páginas xii, 33, 38, 39, 42, 43, 44, 46 e 50]
- [29] C. Reis, “Desenvolvimento modular de uma máquina cnc de 3 eixos,” Master’s thesis, Faculdade de Engenharia da Universidade do Porto, 7 2018. [Citado na página 36]
- [30] “Transmissão mecânica.” Available at <https://www.directindustry.com/pt/cat/transmissao-mecanica-L.html>. (Last accessed in 28/08/2024). [Citado nas páginas xii, 37, 44, 45, 46, 47, 48 e 49]
- [31] J. Seybold, A. Bülau, K. P. Fritz, A. Frank, C. Scherjon, J. Burghartz, and A. Zimmermann, “Miniaturized optical encoder with micro structured encoder disc,” *Applied Sciences (Switzerland)*, vol. 9, 1 2019. [Citado nas páginas xii e 40]

- [32] R. Teja, “Position sensors | types, lvdt, rotary encoder.” Available at <https://www.electronicshub.org/position-sensors/>, 2024. (Last accessed in 26/08/2024). [Citado nas páginas xii, 41 e 42]
- [33] H. Kief and H. Roschiwal, *CNC Handbook*. McGraw Hill LLC, 2012. [Citado na página 49]
- [34] G. Lotti, V. Villani, N. Battilani, and C. Fantuzzi, “New trends in the design of human-machine interaction for cnc machines,” *IFAC-PapersOnLine*, vol. 52, no. 19, pp. 31–36, 2019. 14th IFAC Symposium on Analysis, Design, and Evaluation of Human Machine Systems HMS 2019. [Citado nas páginas xii e 52]
- [35] K. Obrovac, P. Raos, T. Galeta, J. Nižetić, A. Mutka, J. V. Obrovac, and Z. Vrhovski, “A new approach to the design of a cnc machine for making orthotic moulds,” *Tehnicki Vjesnik*, vol. 25, pp. 460–465, 9 2018. [Citado nas páginas 56, 57, 58 e 60]
- [36] P. Smid, *CNC Programming Handbook: A Comprehensive Guide to Practical CNC Programming*. Industrial Press eBooks, Industrial Press, 2003. [Citado nas páginas 57 e 58]
- [37] “Vorum prosthetics & orthotics carvers.” Available at <https://vorum.com/op-carvers/>. (Last accessed in 13/06/2024). [Citado nas páginas xii e 57]
- [38] “Milling machines.” Available at <https://www.rodin4d.com/en/fraiseuse-sx/>. (Last accessed in 17/06/2024). [Citado nas páginas xii, xvii, 60 e 61]
- [39] “Vulcan - computerized numerical control milling machine.” Available at www.sensormedica.com/en/vulcan-cnc/. (Last accessed in 19/06/2024). [Citado nas páginas xii, xvii, 62 e 63]
- [40] R. S. J. and S. Y. C., “Modeling and control of cnc machines using a pc-based open architecture controller,” *Mechatronics*, vol. 5, pp. 401–420, 1995. [Citado nas páginas 64 e 66]
- [41] J. E. Correa, N. Toombs, and P. M. Ferreira, “A modular-architecture controller for cnc systems based on open-source electronics,” *Journal of Manufacturing Systems*, vol. 44, pp. 317–323, 7 2017. [Citado nas páginas 64 e 66]
- [42] “Cnc retrofit kits.” Available at <https://www.machinetoolproducts.com/cnc-retrofit-kits/>. (Last accessed in 29/06/2024). [Citado nas páginas xii, 64 e 65]
- [43] Available at <https://pt.aliexpress.com/w/wholesale-cnc-kit.html>. (Last accessed in 30/06/2024). [Citado nas páginas xii, 65 e 66]

- [44] L. Oliveira, “Percursos de maquinagem 2d: mesa cnc e robô industrial,” Master’s thesis, Faculdade de Engenharia da Universidade do Porto, 6 2018. [Citado na página 67]
- [45] “Cnc software products.” Available at <https://www.mastercam.com/solutions/products/>. (Last accessed in 30/08/2024). [Citado nas páginas xii, 67 e 68]
- [46] “Deskproto software review: Is it worth it?.” Available at <https://mellowpine.com/deskproto-software-review/>. (Last accessed in 31/08/2024). [Citado nas páginas xiii e 68]
- [47] “Universal gcode sender.” Available at <https://universalgcodesender.com/>. (Last accessed in 30/08/2024). [Citado nas páginas xiii e 69]
- [48] “Mach4.” Available at <https://www.machsupport.com/software/mach4/>. (Last accessed in 30/08/2024). [Citado nas páginas xiii e 70]
- [49] J. Veeramony and M. N. O. Zahid, “A customizable controller for 3 axis modular cnc machine,” *Journal of Modern Manufacturing Systems and Technology*, vol. 6, pp. 55–62, 9 2022. [Citado na página 70]
- [50] K. G. S. Rajendra, and C. Keshava, “Retrofitting of cnc bench milling machine with raspberry pi,” *International Research Journal of Engineering and Technology*, vol. 8, 8 2021. [Citado na página 70]
- [51] “What is marlin?.” Available at <https://marlinfw.org/docs/basics/introduction.html#what-is-marlin?> (Last accessed in 31/08/2024). [Citado na página 71]
- [52] “What is grbl?.” Available at <https://www.grbl.org/what-is-grbl>. (Last accessed in 31/08/2024). [Citado na página 71]
- [53] M. Groover, *FUNDAMENTALS OF MODERN MANUFACTURING: Materials, Processes, and Systems*. Wiley, 2009. [Citado nas páginas xiii, 71, 72, 73 e 74]
- [54] P. Almeida, “Programação de robôs para realização de tarefas de maquinação,” Master’s thesis, Instituto Superior de Engenharia do Porto, 11 2016. [Citado nas páginas 71, 72, 73 e 74]
- [55] “Fresas de topo de metal duro - mp series.” Available at <https://www.meetyoucarbide.com/pt/shop/mp-4-flutes-ball-nose-end-mills/>. (Last accessed in 31/08/2024). [Citado nas páginas xiii e 83]
- [56] “Nema 23 bipolar.” Available at <https://www.omc-stepperonline.com/nema-23-bipolar-3nm-425oz-in-3-5a-57x57x114mm-4-wires-cnc-stepper-motor-23hs45-35> (Last accessed in 2/09/2024). [Citado nas páginas xiii, xv, 84, 85, 104 e 155]

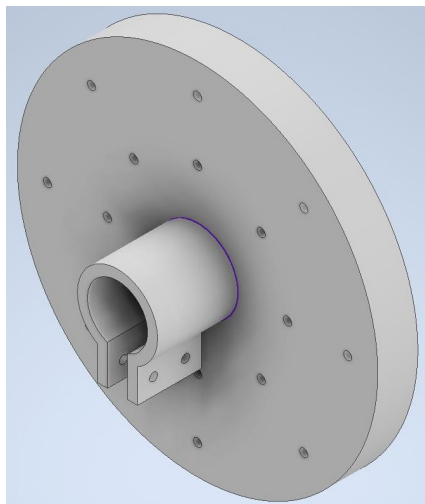
- [57] “Nema 34 bipolar.” Available at <https://www.omc-stepperonline.com/nema-34-cnc-high-torque-stepper-motor-13nm-1841oz-in-5a-86x86x150mm-34hs59-5004s>. (Last accessed in 2/09/2024). [Citado nas páginas xiii, xv, 84, 85, 104 e 156]
- [58] “Electric spindle for stone engraving.” Available at <https://www.tsspindle.com/Product/GDZ12-5-5-5.html>. (Last accessed in 31/08/2024). [Citado nas páginas xiii e 85]
- [59] “Arduino mega 2560 rev3.” Available at <https://docs.arduino.cc/hardware/mega-2560/>. (Last accessed in 2/09/2024). [Citado nas páginas xiii, 87, 107 e 108]
- [60] “Digital stepper driver 2.4-7.2a 20-80vdc for nema 34 stepper motor cnc drive.” Available at <https://www.omc-stepperonline.com/digital-stepper-driver-2-4-7-2a-20-80vdc-for-nema-34-stepper-motor-cnc-drive-dm860I?search=DM860I>. (Last accessed in 2/09/2024). [Citado nas páginas xiii, xiv, xv, 88, 99, 100, 103, 104, 106, 108 e 161]
- [61] “Digital stepper driver 1.0-4.5a 18-50vdc for nema 17, 23, 24 stepper motor.” Available at <https://www.omc-stepperonline.com/digital-stepper-driver-1-0-4-2a-20-50vdc-for-nema-17-23-24-stepper-motor-dm542t?search=DM542T>. (Last accessed in 2/09/2024). [Citado nas páginas xiii, xiv, xv, 88, 103, 104, 105, 106, 107 e 161]
- [62] *YL620 A Inverter Manual PDF*. (Last accessed in 31/08/2024). [Citado nas páginas xiii, xv, 89 e 156]
- [63] “Switching power supply.” Available at <https://www.omc-stepperonline.com/350w-60v-5-9a-115-230v-switching-power-supply-stepper-motor-cnc-router-kits-s>. (Last accessed in 2/09/2024). [Citado nas páginas xiii, xiv, xv, 89, 90, 103, 104, 105 e 158]
- [64] “grbl-mega-5x.” Available at <https://github.com/fra589/grbl-Mega-5X>. (Last accessed in 31/08/2024). [Citado nas páginas xiii e 90]
- [65] “Nema 23 stepper motor.” Available at <https://microcontrollerslab.com/nema-23-stepper-motor-pinoutfeatures-example-arduino/>. (Last accessed in 31/08/2024). [Citado nas páginas xiii, 99 e 100]
- [66] “Nema 34 stepper motor.” Available at <https://microcontrollerslab.com/nema-34-stepper-motor-pinout-features-wiring-interfacing-with-arduino/>. (Last accessed in 31/08/2024). [Citado nas páginas xiii, 100 e 101]
- [67] K. K. G, S. Rajendra, and C. K. K. R, “Retrofitting of cnc bench milling machine with raspberry pi,” *International Research Journal of Engineering and Technology*, 2021. [Citado na página 102]

-
- [68] Available at <https://github-wiki-see.page/m/grbl/grbl/wiki/Configuring-Grbl-v0.9>. (Last accessed in 31/08/2024). [Citado na página 109]
- [69] Available at <https://github-wiki-see.page/m/grbl/grbl/wiki/Interfacing-with-Grbl>. (Last accessed in 31/08/2024). [Citado na página 109]
- [70] “Cpu and pin mapping configuration file part of grbl.” Available at https://github.com/grbl/grbl/blob/master/grbl/cpu_map/cpu_map_atmega2560.h. (Last accessed in 31/08/2024). [Citado na página 110]
- [71] “Product reference manual - arduino® mega 2560 rev3.” Available at <https://docs.arduino.cc/resources/datasheets/A000067-datasheet.pdf>. (Last accessed in 2/09/2024). [Citado nas páginas xv e 157]
- [72] “User’s manual for dm860i - fully digital stepper drive - version 1.0.” Available at <https://www.omc-stepperonline.com/download/DM860I.pdf>. (Last accessed in 31/08/2024). [Citado nas páginas xvii e 160]
- [73] “User manual dm542t (v4.0).” Available at https://www.omc-stepperonline.com/download/DM542T_V4.0.pdf. (Last accessed in 31/08/2024). [Citado nas páginas xvii e 160]
- [74] “Grbl g-code commands list: A guide for cnc users.” Available at https://cncphilosophy.com/grbl-g-code-commands-list/?utm_content=cmp-true. (Last accessed in 1/09/2024). [Citado nas páginas xvii, 163, 164, 165 e 166]
- [75] “Grbl v1.1 commands.” Available at <https://github.com/gnea/grbl/blob/master/doc/markdown/commands.md>. (Last accessed in 1/09/2024). [Citado nas páginas xvii, 164, 165 e 166]

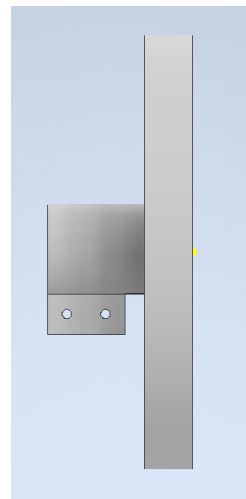
Anexo A

Desenhos CAD

A.1 Desenho CAD da peça de fixação



(a)



(b)

Figura A.1: Desenhos CAD da peça de fixação

A.2 Desenho técnico da peça de fixação

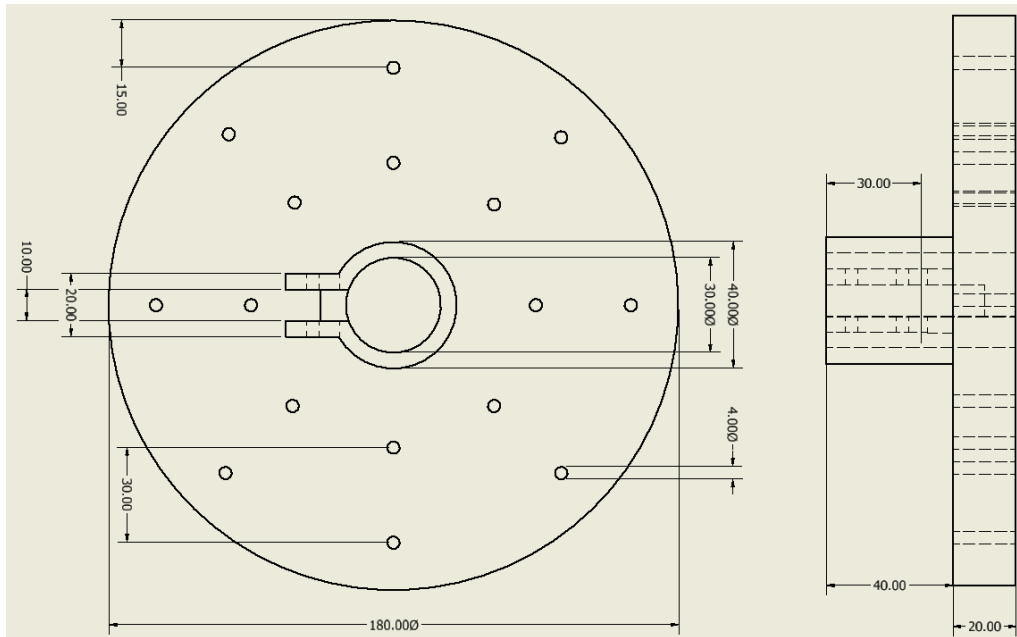


Figura A.2: Desenho técnico da peça de fixação

Anexo B

Especificações do sistema CNC "QRC"

B.1 Dimensões dos motores

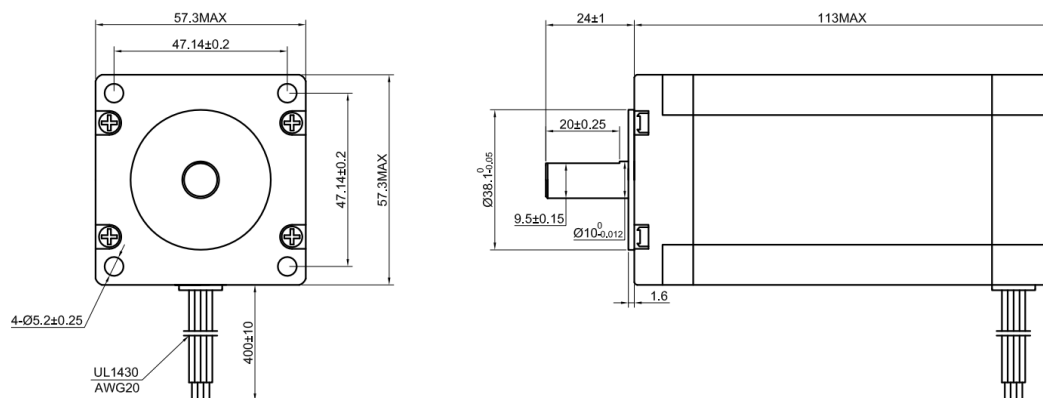


Figura B.1: Dimensões do motor NEMA 23 [56]

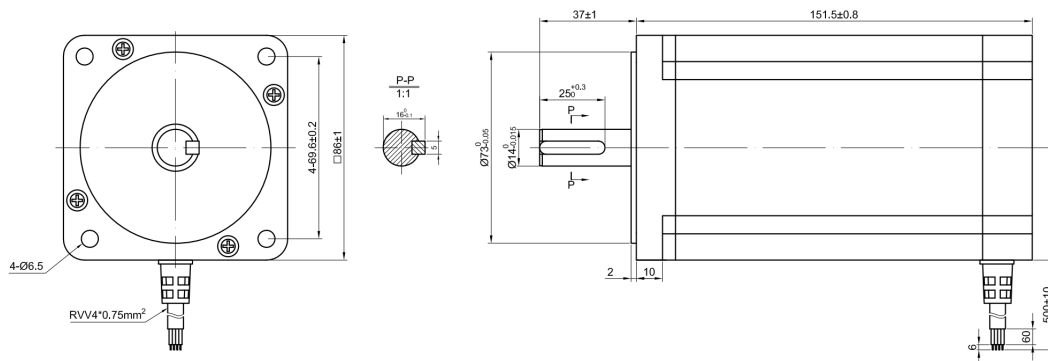


Figura B.2: Dimensões do motor NEMA 34 [57]

B.2 Pinout inversor do Spindle

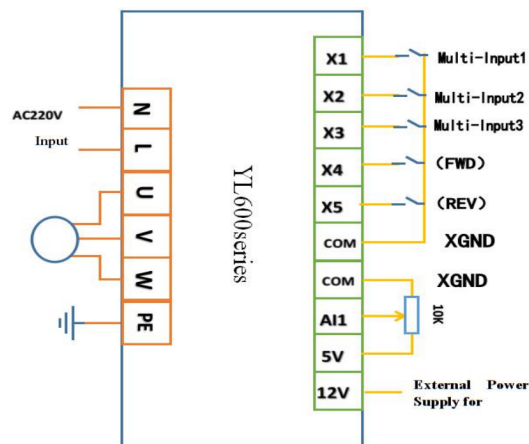


Figura B.3: Esquema de ligações possíveis ao inversor do Spindle YL620-A [62]

B.3 Pinout Arduino Mega 2560 Rev3

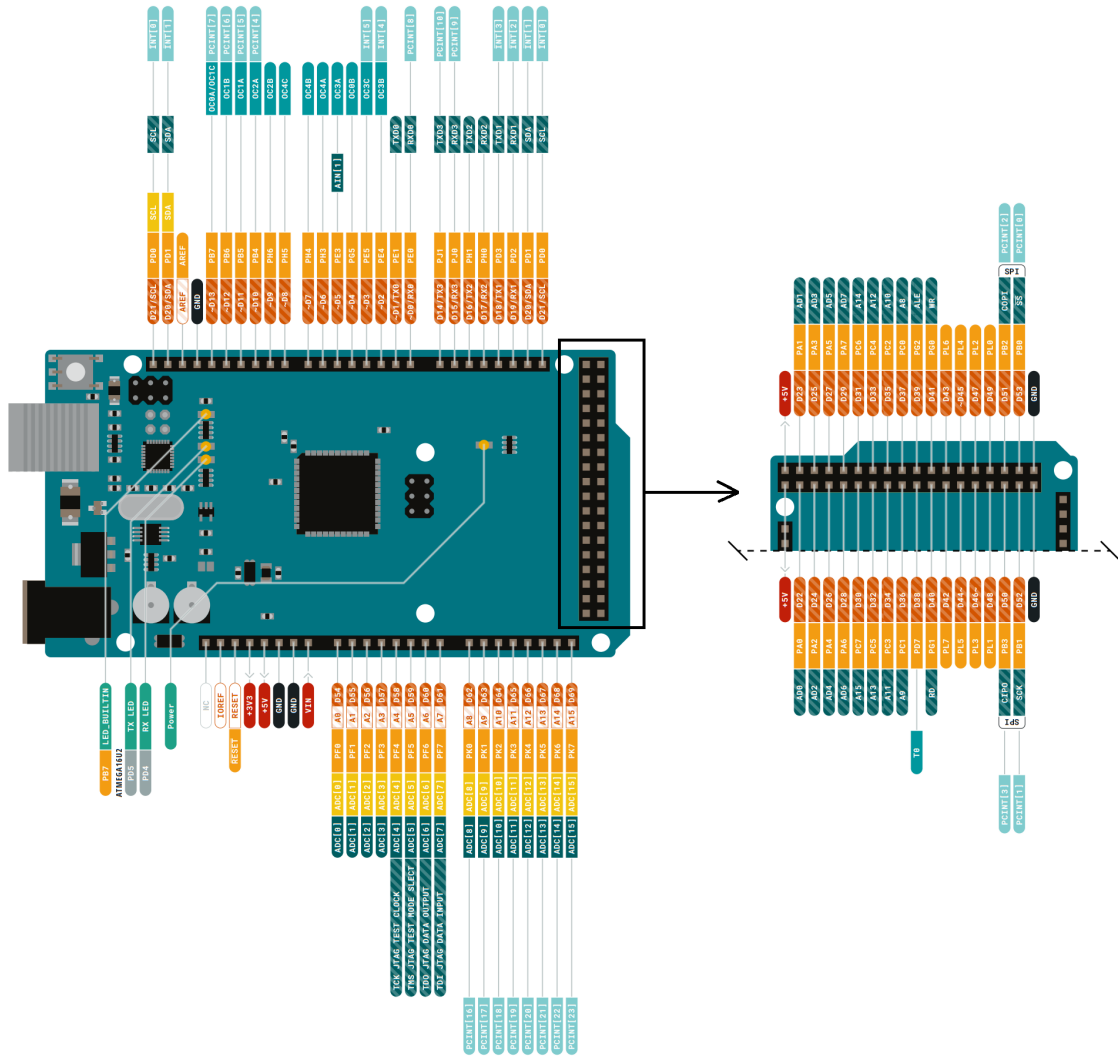


Figura B.4: Pinout Arduino Mega 2560 Rev3 [71]

B.4 Dimensões da fonte de alimentação

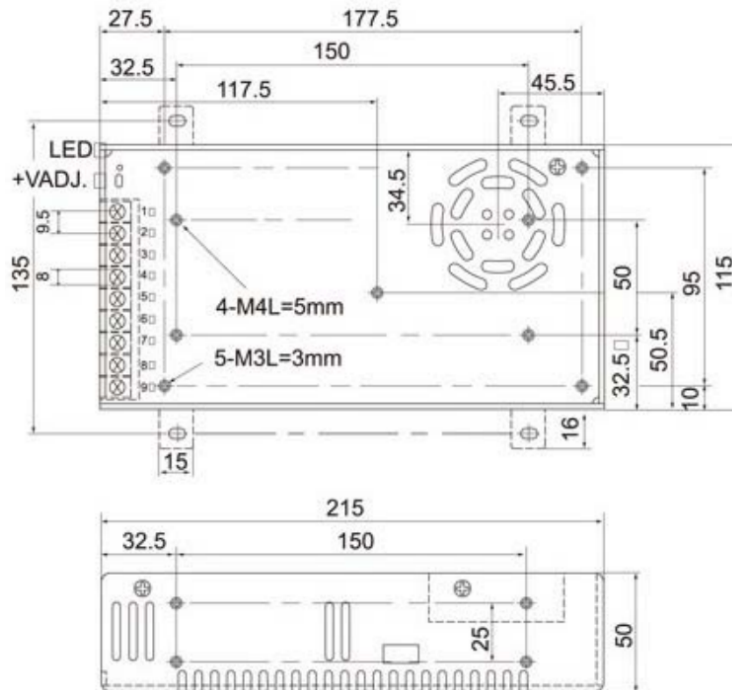


Figura B.5: Dimensões da fonte de alimentação S-350-60 [63]

B.5 Circuito elétrico do sistema CNC "QRC"

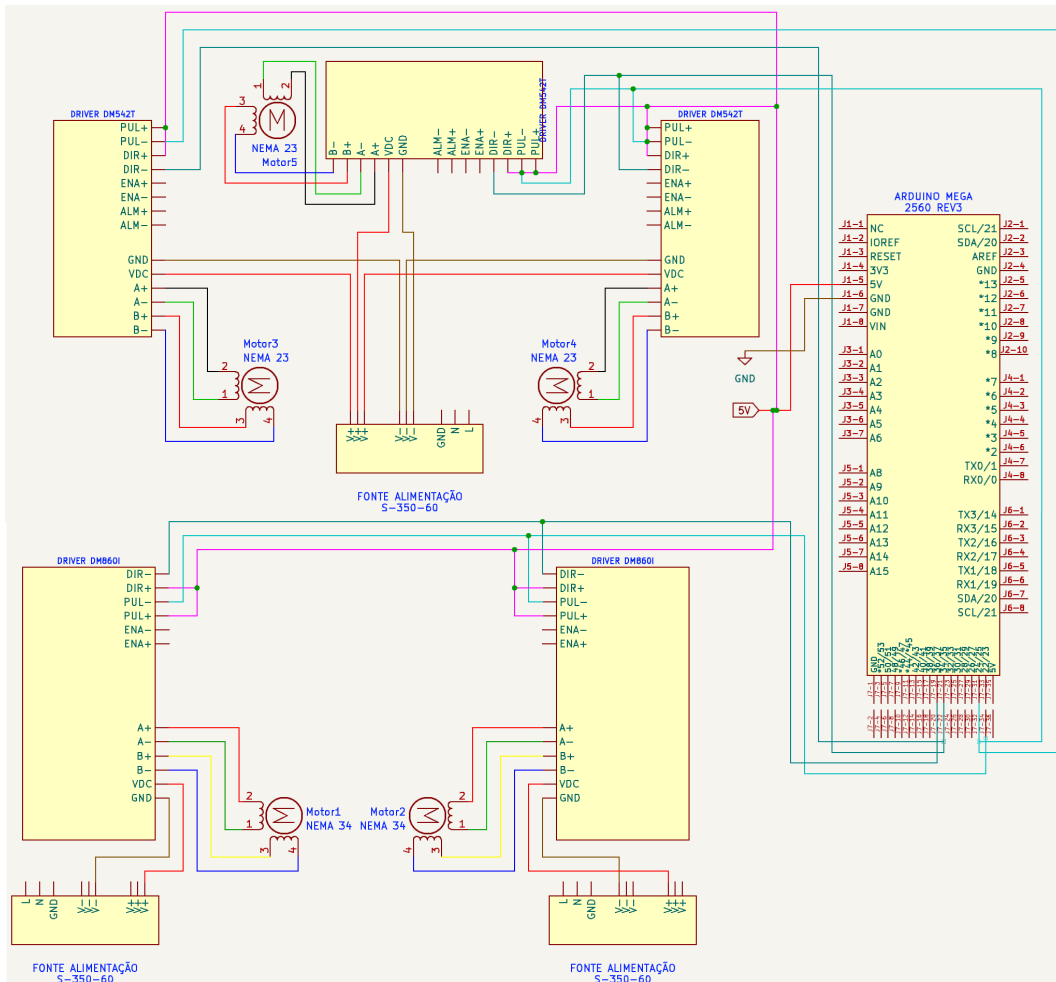


Figura B.6: Circuito elétrico do sistema CNC "QRC"

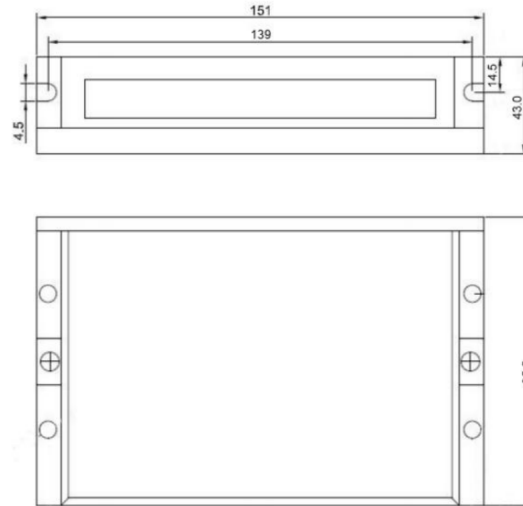
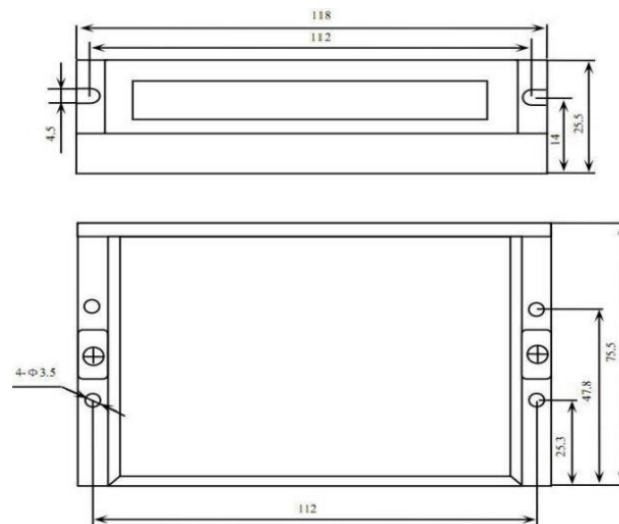
B.6 Resolução dos *drivers*

Tabela B.1: Tabela dos micropassos e passos/revolução do *driver* DM860I [72]

Microstep	Steps/rev.(for 1.8°motor)	SW5	SW6	SW7	SW8
2	400	ON	ON	ON	ON
4	800	OFF	ON	ON	ON
8	1600	ON	OFF	ON	ON
16	3200	OFF	OFF	ON	ON
32	6400	ON	ON	OFF	ON
64	12800	OFF	ON	OFF	ON
128	25600	ON	OFF	OFF	ON
256	51200	OFF	OFF	OFF	ON
5	1000	ON	ON	ON	OFF
10	2000	OFF	ON	ON	OFF
20	4000	ON	OFF	ON	OFF
25	5000	OFF	OFF	ON	OFF
40	8000	ON	ON	OFF	OFF
50	10000	OFF	ON	OFF	OFF
100	20000	ON	OFF	OFF	OFF
200	40000	OFF	OFF	OFF	OFF

Tabela B.2: Tabela dos micropassos e passos/revolução do *driver* DM542T [73]

Microstep	Steps/rev.(for 1.8°motor)	SW5	SW6	SW7	SW8
1	200	ON	ON	ON	ON
2	400	OFF	ON	ON	ON
4	800	ON	OFF	ON	ON
8	1600	OFF	OFF	ON	ON
16	3200	ON	ON	OFF	ON
32	6400	OFF	ON	OFF	ON
64	12800	ON	OFF	OFF	ON
128	25600	OFF	OFF	OFF	ON
5	1000	ON	ON	ON	OFF
10	2000	OFF	ON	ON	OFF
20	4000	ON	OFF	ON	OFF
25	5000	OFF	OFF	ON	OFF
40	8000	ON	ON	OFF	OFF
50	10000	OFF	ON	OFF	OFF
100	20000	ON	OFF	OFF	OFF
125	25000	OFF	OFF	OFF	OFF

B.7 Dimensões dos *drivers*Figura B.7: Dimensões do *driver* DM860I [60]Figura B.8: Dimensões do *driver* DM542T [61]

Anexo C

G-code e comandos do sistema Grbl

C.1 G-code suportado pelo Grbl

O G-code utilizado no *firmware* Grbl desempenham um papel fundamental na operação das máquinas CNC, permitindo um controlo preciso de várias funções que permitem a maquinação de peças. O G-code suportado pelo Grbl abrange diversos fatores do processo de maquinação, como os modos de movimento que determinam como a ferramenta se desloca relativamente à peça, modos de taxa de avanço, que controlam a velocidade com que a ferramenta se move e modos de distância, que especificam se as coordenadas se encontram em absolutas ou relativas. As **Figuras** C.1 e C.2 mostram os comandos mais utilizados neste *firmware* [74].

Tabela C.1: G-code suportado pelo *firmware* Grbl: comandos modais [74] [75]

Grupos modais	Palavras de comando e respetivo significado	
Modo de movimento	G0	Posicionamento rápido
	G1	Interpolação linear
	G2	Interpolação circular no sentido horário
	G3	Interpolação circular no sentido anti-horário
	G38.2, G38.3, G38.4, G38.5	Sondagem em direção à peça de trabalho
	G80	Cancelamento ciclo pré-programado
Seleção do sistema de coordenadas	G54, G55, G56, G57, G58, G59	Sistema de coordenadas de trabalho (sistema 1 a 6, respetivamente)
Seleção do plano	G17	Plano XY
	G18	Plano ZX
	G19	Plano YZ
Modo de unidades	G20	Polegadas (inch)
	G21	Milímetros (mm)
Compensação do raio da ferramenta	G40	Cancelar compensação do raio da ferramenta
Offset altura da ferramenta	G43.1	Definir compensação de altura da ferramenta de modo dinâmico
	G49	Cancelar compensação de altura da ferramenta
Modo de distância	G90	Posicionamento absoluto
	G91	Posicionamento incremental
Modo de distância arco IJK	G91.1	Definir os valores IJK incrementais
Modo de taxa de avanço	G93	Modo de tempo invertido
	G94	Modo de unidades por minuto
Modo programação	M0	Paragem do programa
	M1	Paragem opcional do programa
	M2	Fim do programa
	M30	Fim do programa e voltar ao início
Estado do <i>Spindle</i>	M3	Ligar <i>spindle</i> no sentido horário
	M4	Ligar <i>spindle</i> no sentido anti-horário
	M5	Desligar <i>spindle</i>
Estado da refrigeração	M7, M8, M9	Ligar refrigeração névoa e jato e desligar refrigeração (respetivamente)
Controlo <i>override</i>	M56	Controlar anulação do movimento de estacionar

Tabela C.2: G-code suportado pelo *firmware* Grbl: comandos não modais [74] [75]

Grupo não modal	Palavras de comando e respetivo significado	
Comandos não modais	G4	<i>Dwell</i> (pausa)
	G10L2	Definir sistema de coordenadas de trabalho
	G10L20	Definir sistema de coordenadas de trabalho (relativas)
	G28	Regressar à posição de referência
	G28.1	Definir posição de referência
	G30	Regressar à segunda posição de referência
	G30.1	Definir segunda posição de referência
	G53	Movimento nas coordenadas da máquina
	G92	Definir deslocamento do sistema de coordenadas de trabalho
	G92.1	Cancelar deslocamento do sistema de coordenadas de trabalho

C.2 Comandos próprios do sistema Grbl

O Grbl, além de suportar uma variedade de G-code, tem um conjunto específico de comandos do sistema, conhecidos como comandos '\$', que permitem aos utilizadores das máquinas CNC configurar diversos parâmetros, realizar ciclos de referência ou aceder a algumas funções de controlo da máquina. Na **Figura C.3** é possível verificar o sistema de comandos \$ do Grbl.

Tabela C.3: Sistema de comandos \$ específico do Grbl [74] [75]

Comando	Significado do comando
\$\$	Ver configurações do Grbl
\$x=valor	Configurações parâmetros do Grbl
\$#	Ver parâmetros do G-code
\$G	Ver estado do interpretador do G-code
\$I	Ver informação de compilação
\$N	Ver bloco de inicialização
\$Nx=linha	Guardar bloco de inicialização
\$C	Verificar modo do G-code
\$X	Desbloquear modo de alarme
\$H	Executar ciclo <i>Home</i>
\$Jx=linha	Executar movimento de <i>jogging</i>
\$RST=\$, \$RST=#, \$RST=*	Restaurar configurações e cados do Grbl
\$SLP	Modo de suspensão

Além disso, o Grbl tem, ainda, comandos em tempo real ASCII que podem ser digitados diretamente pelo utilizador no teclado ou interface do utilizador enquanto a máquina CNC está em operação. Estes comandos são processados instantaneamente pelo Grbl, independentemente da execução do programa no qual a máquina se encontrar, permitindo ao operador ter controlo imediato de algumas funções básicas da máquina e são, especialmente, úteis em situações onde é necessário agir rapidamente. A **Figura C.4** permite observar os comandos ASCII do Grbl.

Tabela C.4: Sistema de comandos \$ ASCII, específico do Grbl [74]
[75]

Comandos ASCII	Significado do comando
0x18 (ctrl-x)	<i>Reset do firmware</i>
?	Consultar relatório de estado
~	Iniciar ciclo ou continuar
!	Pausa de avanço

Anexo D

Código utilizado no *software* Arduino IDE

D.1 Código utilizado para testar os motores

D.1.1 Código para testar motores NEMA 23

```
1 /* Definicao dos pinos de controle */
2 #define STEP_PIN 24 // Pino de passo
3 #define DIR_PIN 35 // Pino de direcao
4 #define ENABLE_PIN 28 // Pino de habilitacao do driver
5
6 void setup() {
7     /* Configuracao dos pinos */
8     pinMode(STEP_PIN, OUTPUT);
9     pinMode(DIR_PIN, OUTPUT);
10    pinMode(ENABLE_PIN, OUTPUT);
11
12    /* Habilitacao do driver */
13    digitalWrite(ENABLE_PIN, LOW);
14 }
15
16 void loop() {
17
18    /* Girar o motor numa direcao */
```

```

19  digitalWrite(DIR_PIN, HIGH); // Definicao da direcao para HIGH
20  for (int i = 0; i < 600; i++) { // Numero de passos para girar
    numa direcao
21    digitalWrite(STEP_PIN, HIGH);
22    delayMicroseconds(5000); // Tempo do pulso para controlar a
    velocidade
23    digitalWrite(STEP_PIN, LOW);
24    delayMicroseconds(5000); // Intervalo entre os pulsos
25  }
26
27  delay(5000); // Pausa de 5 segundos antes de inverter a direcao
    do motor
28
29  /* Girar o motor na direcao oposta */
30  digitalWrite(DIR_PIN, LOW); // Define a direcao para LOW (
    inverte o sentido)
31  for (int i = 0; i < 600; i++) { // Numero de passos para girar
    na direcao oposta
32    digitalWrite(STEP_PIN, HIGH);
33    delayMicroseconds(5000); // Tempo do pulso para controlar a
    velocidade
34    digitalWrite(STEP_PIN, LOW);
35    delayMicroseconds(5000); // Intervalo entre os pulsos
36  }
37
38  delay(5000); // Pausa de 5 segundos antes de inverter a direcao
    novamente
39  }

```

D.1.2 Código para testar motores NEMA 34

```

1  /* Definicao dos pinos de controlo */
2
3  #define STEP_PIN 22 // Pino de passo
4  #define DIR_PIN 37 // Pino de direcao
5  #define ENABLE_PIN 26 // Pino de habilitacao
6
7  void setup() {
8    /* Configuracao dos pinos */
9    pinMode(STEP_PIN, OUTPUT);
10   pinMode(DIR_PIN, OUTPUT);
11   pinMode(ENABLE_PIN, OUTPUT);
12
13   /* Habilitacao do driver */
14   digitalWrite(ENABLE_PIN, LOW);
15 }
16

```

```
17 void loop() {
18
19     /* Girar o motor numa direcao */
20     digitalWrite(DIR_PIN, HIGH); // Definicao da direcao para HIGH
21     for (int i = 0; i < 600; i++) { // Numero de passos para girar
        numa direcao
22         digitalWrite(STEP_PIN, HIGH);
23         delayMicroseconds(5000); // Tempo do pulso para controlar a
        velocidade
24         digitalWrite(STEP_PIN, LOW);
25         delayMicroseconds(5000); // Intervalo entre os pulsos
26     }
27
28     delay(5000); // Pausa de 5 segundos antes de inverter a direcao
        do motor
29
30     /* Girar o motor na direcao oposta */
31     digitalWrite(DIR_PIN, LOW); // Define a direcao para LOW (
        inverte o sentido)
32     for (int i = 0; i < 600; i++) { // Numero de passos para girar
        na direcao oposta
33         digitalWrite(STEP_PIN, HIGH);
34         delayMicroseconds(5000); // Tempo do pulso para controlar a
        velocidade
35         digitalWrite(STEP_PIN, LOW);
36         delayMicroseconds(5000); // Intervalo entre os pulsos
37     }
38
39     delay(5000); // Pausa de 5 segundos antes de inverter a direcao
        novamente
40 }
```

D.2 Mapeamento de pinos do Arduino para o Grbl

D.2.1 Ficheiro cpu_map.h

```
1 #ifndef cpu_map_h
2 #define cpu_map_h
3
4 /* Mapeamento dos pinos do Arduino para o GRBL */
5 #ifdef CPU_MAP_2560_QRC
6     #include "nuts_bolts.h"
7
8     /* Definicoes da porta serial */
9     #define SERIAL_RX USART0_RX_vect
10    #define SERIAL_UDRE USART0_UDRE_vect
```

```
11
12 /* Definicao dos pinos de pulso de passo (STEP) para cada eixo */
13 #define X1_STEP_PIN      22 // Define pino digital 22 para STEP
    do eixo X1
14 #define X2_STEP_PIN      22 // Define pino digital 22 para STEP
    do eixo X2
15 #define Z_STEP_PIN       24 // Define pino digital 24 para STEP
    do eixo Z
16 #define A1_STEP_PIN      25 // Define pino digital 25 para STEP
    do eixo A1
17 #define A2_STEP_PIN      25 // Define pino digital 25 para STEP
    do eixo A2
18
19 /* Definicao dos pinos de direcao (DIRECTION) para cada eixo */
20 #define X1_DIRECTION_PIN 37 // Define pino digital 37 para DIR do
    eixo X1
21 #define X2_DIRECTION_PIN 37 // Define pino digital 37 para DIR do
    eixo X2
22 #define Z_DIRECTION_PIN  35 // Define pino digital 35 para DIR do
    eixo Z
23 #define A1_DIRECTION_PIN 34 // Define pino digital 34 para DIR do
    eixo A1
24 #define A2_DIRECTION_PIN 34 // Define pino digital 34 para DIR do
    eixo A2
25
26 /* Definicao dos pinos de habilitacao (ENABLE) para cada eixo */
27 #define X1_ENABLE_PIN    32 // Define pino digital 32 para ENABLE
    do eixo X1
28 #define X2_ENABLE_PIN    33 // Define pino digital 33 para ENABLE
    do eixo X2
29 #define Z_ENABLE_PIN     34 // Define pino digital 34 para ENABLE
    do eixo Z
30 #define A1_ENABLE_PIN    35 // Define pino digital 35 para ENABLE
    do eixo A1
31 #define A2_ENABLE_PIN    36 // Define pino digital 36 para ENABLE
    do eixo A2
32
33 /* Definicao dos pinos de fim de curso (LIMIT) para cada eixo */
34 // #define X1_MIN_PIN      3 // Pino digital 3 para fim de
    curso minimo do eixo X1
35 // #define X2_MIN_PIN     14 // Pino digital 14 para fim de
    curso minimo do eixo X2
36 // #define Z_MIN_PIN      18 // Pino digital 18 para fim de
    curso minimo do eixo Z
37 // #define A1_MIN_PIN     14 // Pino digital 14 para fim de
    curso minimo do eixo A1
38 // #define A2_MIN_PIN     14 // Pino digital 14 para fim de
    curso minimo do eixo A2
39
```

```
40 // #define X1_MAX_PIN      3 // Pino digital 3 para fim de
    curso maximo do eixo X1
41 // #define X2_MAX_PIN      14 // Pino digital 14 para fim de
    curso maximo do eixo X2
42 // #define Z_MAX_PIN       18 // Pino digital 18 para fim de
    curso maximo do eixo Z
43 // #define A1_MAX_PIN      14 // Pino digital 14 para fim de
    curso maximo do eixo A1
44 // #define A2_MAX_PIN      14 // Pino digital 14 para fim de
    curso maximo do eixo A2
45
46 /* Definicao dos pinos da sonda de toque */
47 // #define PROBE_PIN        7 // Comentado porque nao se utiliza
    sonda
48
49 /* Definicao dos pinos de refrigeracao e neblina */
50 // #define COOLANT_FLOOD_PIN 8 // Comentado porque nao se utiliza
    refrigeracao
51 // #define COOLANT_MIST_PIN 9 // Comentado porque nao se utiliza
    neblina
52
53 /* Definicao dos pinos para o spindle */
54 // #define SPINDLE_ENABLE_PIN 10 // Comentado porque o spindle
    funciona manualmente
55 // #define SPINDLE_PWM_PIN   11 // Comentado porque o spindle
    funciona manualmente
56
57 #endif // CPU_MAP_2560_QRC
58
59 #endif // cpu_map_h
```

D.2.2 Ficheiro config.h

```
1 #ifndef config_h
2 #define config_h
3
4 #include "grbl.h" // Para compatibilidade com Arduino IDE
5
6 /* Definicao do serial baud rate */
7 #define BAUD_RATE 115200
8
9 /* Definicao do mapa de pinos CPU pin map e das definicoes padrao
    */
10 #define CPU_MAP_2560_QRC // Utilizacao do mapeamento de pinos
    definido no ficheiro cpu_map.h
11
12 /* Definicao dos eixos */
```

```
13 #define N_AXIS 5 // Numero de eixos
14 #define N_AXIS_LINEAR 3 // Numero de eixos lineares (tipicamente X
    , Y e Z)
15
16 /* Definicao de indices e nomes dos eixos */
17 #define AXIS_1 0 // Indice do eixo 1
18 #define AXIS_1_NAME 'X1' // Nome do eixo 1
19 #define AXIS_2 1 // Indice do eixo 2
20 #define AXIS_2_NAME 'X2' // Nome do eixo 2
21 #define AXIS_3 2 // Indice do eixo 3
22 #define AXIS_3_NAME 'Z' // Nome do eixo 3
23 #define AXIS_4 3 // Indice do eixo 4
24 #define AXIS_4_NAME 'A1' // Nome do eixo 4
25 #define AXIS_5 4 // Indice do eixo 5
26 #define AXIS_5_NAME 'A2' // Nome do eixo 5
27
28 /* Definicao do ciclo home */
29 // #define HOMING_CYCLE_0 (1<<AXIS_3) // Comentado porque nao se
    utiliza ciclo home
30 // #define HOMING_CYCLE_1 ((1<<AXIS_1)|(1<<AXIS_2)) // Comentado
    porque nao se utiliza ciclo home
31 // #define HOMING_CYCLE_2 (1<<AXIS_4) // Comentado porque nao se
    utiliza ciclo home
32 // #define HOMING_CYCLE_3 (1<<AXIS_5) // Comentado porque nao se
    utiliza ciclo home
33
34 /* Numero de ciclos de homing realizados apos o deslocamento
    inicial para evitar overshoot */
35 // #define N_HOMING_LOCATE_CYCLE 1 // Comentado porque nao se
    utiliza ciclo home
36
37 // #define HOMING_SINGLE_AXIS_COMMANDS // Comentado porque nao se
    utiliza ciclo home
38
39 /* Configuracoes do Spindle e PWM */
40 // #define SPINDLE_PWM_ON_D8 // Comentado porque spindle e
    controlado manualmente
41 // #define SPINDLE_PWM_ON_D6 // Comentado porque spindle e
    controlado manualmente
42 // #define SPINDLE_PWM_ON_D9 // Comentado porque spindle e
    controlado manualmente
43
44 // #undef SEPARATE_SPINDLE_LASER_PIN // Comentado porque spindle e
    controlado manualmente
45 // #undef USE_OUTPUT_PWM // Comentado porque spindle e controlado
    manualmente
46
47 /* Outras configuracoes do sistema */
48
```

```
49 /* Desabilitar travamento do homing ao inicializar */
50 #undef HOMING_INIT_LOCK
51
52 /* Verificacao dos limites ao inicializar */
53 #define CHECK_LIMITS_AT_INIT
54
55 /* Alinhamento dos ciclos de homing para cada eixo */
56 // #define HOMING_AXIS_SEARCH_SCALAR 1.5 // Comentado porque nao
    se utiliza ciclo home
57 // #define HOMING_AXIS_LOCATE_SCALAR 10.0 // Comentado porque nao
    se utiliza ciclo home
58
59 /* Numero de pontos decimais para valores de coordenadas, taxas e
    configuracoes */
60 #define N_DECIMAL_COORDVALUE_INCH 4
61 #define N_DECIMAL_COORDVALUE_MM 3
62 #define N_DECIMAL_RATEVALUE_INCH 1
63 #define N_DECIMAL_RATEVALUE_MM 0
64 #define N_DECIMAL_SETTINGVALUE 3
65 #define N_DECIMAL_RPMVALUE 0
66
67 /* Velocidade minima de juncao do planner */
68 #define MINIMUM_JUNCTION_SPEED 0.0 // (mm/min)
69
70 /* Taxa de alimentacao minima que o planner permite */
71 #define MINIMUM_FEED_RATE 1.0 // (mm/min)
72
73 /* Habilitar suavizacao adaptativa de passos em multiplos eixos (
    AMASS) */
74 #define ADAPTIVE_MULTI_AXIS_STEP_SMOOTHING
75
76 /* Definicao do eixo que aplica o offset de comprimento da
    ferramenta */
77 #define TOOL_LENGTH_OFFSET_AXIS AXIS_3 // Eixo Z
78
79 /* Sincronizacao do buffer durante gravacao no EEPROM */
80 #define FORCE_BUFFER_SYNC_DURING_EEPROM_WRITE
81
82 /* Sincronizacao do buffer durante mudancas de WCO */
83 #define FORCE_BUFFER_SYNC_DURING_WCO_CHANGE
84
85 /* Velocidade dos ticks de aceleracao */
86 #define ACCELERATION_TICKS_PER_SECOND 100
87
88 /* Definicao da duracao da inatividade antes de entrar no modo de
    suspensao */
89 #define SLEEP_DURATION 5.0 // Duracao em segundos (0.25 - 61.0).
90
91 /* Desabilitar laser durante pausa */
```

```
92 // #define DISABLE_LASER_DURING_HOLD // Comentado porque nao se
    utiliza laser
93
94 #endif // config_h
```

D.2.3 Ficheiro stepper.h

```
1 #ifndef stepper_h
2 #define stepper_h
3
4 /* Definicao tamanho do buffer de segmento */
5 #ifndef SEGMENT_BUFFER_SIZE
6 #define SEGMENT_BUFFER_SIZE 10
7 #endif
8
9 /* Inicializacao e configuracao do subsistema de motores de passo
    */
10 void stepper_init();
11
12 /* Habilitacao dos motores de passo */
13 void st_wake_up(); // ciclo so inicia quando se chama controlo do
    movimento ou comando em tempo real
14
15 /* Desabilitacao imediata dos motores de passo */
16 void st_go_idle();
17
18 /* Gerar as mascaras de inversao das portas de passo e direcao */
19 void st_generate_step_dir_invert_masks();
20
21 /* Reset das variaveis do subsistema de motores de passo */
22 void st_reset();
23
24 /* Recarregar o buffer de segmentos de passo */
25 void st_prep_buffer(); // chamado continuamente pelo sistema de
    execucao em tempo real
26
27 /* Chamado pelo planner_recalculate() quando o bloco em execucao e
    atualizado pelo novo plano */
28 void st_update_plan_block_parameters();
29
30 /* Retorno da taxa de execucao em tempo real */
31 float st_get_realtime_rate(); // retorna se estiver habilitada em
    config.h
32
33 #endif
```

D.2.4 Ficheiro spindle_control.h

```
1 #ifndef spindle_control_h
2 #define spindle_control_h
3
4 #define SPINDLE_NO_SYNC false
5 #define SPINDLE_FORCE_SYNC true
6
7 #define SPINDLE_STATE_DISABLE 0 // Desativado porque spindle e
   controlado manualmente
8 #define SPINDLE_STATE_CW      bit(0) // Rotacao no sentido
   horario
9 #define SPINDLE_STATE_CCW     bit(1) // Rotacao no sentido anti-
   horario
10
11 /* Inicializacao dos pinos do spindle e o hardware PWM */
12 void spindle_init();
13
14 /* Retorna o estado atual do spindle */
15 uint8_t spindle_get_state();
16
17 /* Sincronizacao do estado do spindle com o buffer de movimento */
18 void spindle_sync(uint8_t state, float rpm);
19
20 /* Definicao do estado de funcionamento do spindle com direcao,
   habilitacao e PWM */
21 void spindle_set_state(uint8_t state, float rpm);
22
23 /* Definicao rapida do PWM do spindle para o ISR do stepper */
24 void spindle_set_speed(uint16_t pwm_value);
25
26 /* Calculo do valor do registador PWM especifico do Mega2560 para
   o RPM dado */
27 uint16_t spindle_compute_pwm_value(float rpm);
28
29 /* Funcao para parar o spindle */
30 void spindle_stop();
31
32 #endif
```

D.2.5 Ficheiro spindle_control.c

```
1 #include "grbl.h"
2
3 /* Funcao vazia de inicializacao do spindle */
```

```

4 void spindle_init() {}
5
6 / Retorna o estado do spindle sempre como desativado */
7 uint8_t spindle_get_state() {
8     return SPINDLE_STATE_DISABLE;
9 }
10
11 /* Funcao vazia para parar o spindle */
12 void spindle_stop() {}
13
14 /* Funcao vazia para definir a velocidade do spindle */
15 void spindle_set_speed(uint16_t pwm_value) {}
16
17 /* Funcao vazia para calcular o valor de PWM do spindle */
18 uint16_t spindle_compute_pwm_value(float rpm) {
19     return 0; // Retorna zero para indicar spindle desligado
20 }
21
22 /* Funcao para definir o estado do spindle sem fazer nada */
23 void spindle_set_state(uint8_t state, float rpm) {}
24
25 /* Sincroniza o estado do spindle sem fazer nada */
26 void spindle_sync(uint8_t state, float rpm) {}

```

D.2.6 Ficheiro probe.h

```

1 #ifndef probe_h
2 #define probe_h
3
4 #define PROBE_OFF      0 // Sonda desabilitada ou nao utilizada
5 #define PROBE_ACTIVE  0 // Sem funcionalidade, sempre desabilitado
6
7 /* Rotina de inicializacao do pino de probing */
8 void probe_init(); // funcao vazia porque nao se utiliza sonda
9
10 /* Configuracao da mascara de inversao do pino de probe de acordo
    com a configuracao */
11 void probe_configure_invert_mask(uint8_t is_probe_away); // funcao
    vazia porque nao se utiliza sonda
12
13 /* Retorna o estado do pino de probe */
14 uint8_t probe_get_state();
15
16 /* Monitorizacao o estado do pino de probe e registra a posicao do
    sistema quando detectado */
17 void probe_state_monitor(); // funcao vazia porque nao se utiliza
    sonda

```

```
18
19 #endif
```

D.2.7 Ficheiro coolant_control.h

```
1 #ifndef coolant_control_h
2 #define coolant_control_h
3
4 /* Definicao da sincronizacao para controlo da refrigeracao */
5 #define COOLANT_NO_SYNC      false
6 #define COOLANT_FORCE_SYNC  true
7
8 /* Definicao do estado da refrigeracao
9 #define COOLANT_STATE_DISABLE 0 // Sempre desabilitado
10 #define COOLANT_STATE_FLOOD  0 // Sem funcionalidade
11 #define COOLANT_STATE_MIST   0 // Sem funcionalidade
12
13 // Inicializa os pinos de controle da refrigeracao */
14 // Esta funcao esta presente apenas para manter a compatibilidade
    com o restante do codigo
15 void coolant_init();
16
17 // Retorna o estado atual da refrigeracao */
18 uint8_t coolant_get_state();
19
20 // Desabilita imediatamente os pinos de refrigeracao */
21 // Funcao sem implementacao porque o controle de refrigeracao nao
    esta em uso.
22 void coolant_stop();
23
24 // Definicao dos pinos de refrigeracao de acordo com o estado
    especificado */
25 // Funcao sem implementacao porque nao se utiliza o modulo
    refrigeracao
26 void coolant_set_state(uint8_t mode);
27
28 // Funcao de sincronizacao do parser G-code para definicao dos
    estados do refrigeracao */
29 // Funcao sem implementacao porque nao se utiliza o modulo
    refrigeracao
30 void coolant_sync(uint8_t mode);
31
32 #endif
```

D.2.8 Ficheiro coolant_control.c

```
1 #include "grbl.h"
2
3 /* Inicializacao do modulo de refrigeracao */
4 void coolant_init() {
5     // Funcao sem implementacao porque nao se utiliza refrigeracao
6 }
7
8 /* Retorna o estado atual da refrigeracao */
9 uint8_t coolant_get_state() {
10     return COOLANT_STATE_DISABLE; // Retorna sempre como
        desabilitado.
11 }
12
13 /* Para a refrigeracao */
14 void coolant_stop() {
15     // Funcao sem implementacao porque nao se utiliza o modulo
        refrigeracao
16 }
17
18 /* Definicio do estado da refrigeracao */
19 void coolant_set_state(uint8_t mode) {
20     // Funcao sem implementacao porque nao se utiliza o modulo
        refrigeracao
21 }
22
23 /* Sincronizacao do estado da refrigeracao */
24 void coolant_sync(uint8_t mode) {
25     // Funcao sem implementacao porque nao se utiliza o modulo
        refrigeracao
26 }
```

D.2.9 Ficheiro limits.h

```
1 #ifndef limits_h
2 #define limits_h
3
4 /* Inicializacao do modulo switch */
5 void limits_init();
6
7 /* Retorna o estado do switch como uma variavel bitwise uint8 */
8 uint8_t limits_get_state();
9
```

```
10 /* Verificacao da ocorrencia de violacoes de limites de software
    */
11 void limits_soft_check(float *target);
12
13 /* Funcao vazia para homing porque o homing esta desativado */
14 static inline void limits_go_home(uint8_t cycle_mask) {
15 }
16
17 #endif
```
